NETWORK DIGITAL EVIDENCES CENTRALIZATION BY USING HONEYNET ARCHITECTURE

MOHAMMED ABBAS ALAMEEN SALEH

A project report submitted in partial fulfillment of the requirements for the award of the degree of Master of Computer Science (Information Security)

UNIVERSITI TEKNOLOGI MALAYSIA

OCTOBER 2009

ACKNOWLEDGEMENTS

I would like to express the deepest appreciation to my supervisor Prof. Azizah Abdul Manaf , who has the attitude and the substance of a genius: she continually and convincingly conveyed a spirit of adventure in regard to research, and an excitement in regard to teaching. Without her guidance and persistent help this possible would not have been possible.

I am deeply indebted to all of my lecturers and examiners Dr. Rabia Ahmed; program coordinator, Dr. Kamarulnizam Abu Bakar, and Assoc Prof Zailani for their insights, guidances and encouragements that helped me in all the time of research for and writing of this thesis to improve and enhance the project value and quality.

I am also indebted to Universiti Teknologi Malaysia (UTM) and Center for Advanced Software Engineering for offering this great opportunity to study and learn one of the computer science field ; Information Security. Also deserve special thanks for their assistance in supplying the relevant literatures.

Lastly, I would like to express my greatest appreciation to my family members; my father, my mother, my brother, and my sister who were always with me by their motivations and supporting.

ABSTRACT

The main purpose of this project is to collect and centralize network's data which might be used as digital evidences for the sake the investigation. This project focuses on network rather than a computer because of the reliability of collected and centralized digital evidences. However, a computer is considered not reliable anymore because of its data that can be tampered with by an attacker after conducting the crime. Therefore, finding another place rather than a computer is the first contribution of this project in order to find out its advantages and disadvantages which related to the security and integrity. The key solution in this case is using Honeynets which guarantee reliable digital evidences. Honeywall is the most important component of Honeynet Architecture which is used as a network gateway in hidden manner. However, Honeywall stealthy is achieved from working under Bridging Mode of networking; which is not assigned Internet Protocol and also keeps it to be undetectable from the outside world. Several tools are installed and set up inside Honeywall in order to achieve project aim. Some of these tools are Snort application, Sebek Sever/ Client Architecture, and Log Server Architecture. Snort application used in this project to collect and then centralize the network data into data base. These data is comprehensive all both; encrypted and unencrypted data. Sebek Sever/ Client Architecture used here to record key loggers have done under encrypted protocols such as Secure Shell (SSH) and then log these recorded data into the data base. The functionality of Log Server is to record what happened inside Servers like current status of the servers processes registered with time and last accesses, and errors and etc. The second contribution of this project is making a comparison among three types of Honeynets in terms of security, time, and cost of network evidences. The final objective to produce guidelines which guide and govern network evidences collection and centralization processes and procedures.

ABSTRAK

Tujuan utama kajian ini adalah untuk memusat dan mengumpul data dari rangkaian yang mana ianya boleh digunakan sebagai bahan bukti dalam bentuk digital untuk tujuan siasatan. Pemusatan dan pengumpulan bukti digital pada peringkat rangkaian adalah lebih dipercayai berbanding komputer. Ini disebabkan data-data yang terdapat pada sesebuah komputer itu lebih mudah dan senang diubah oleh seseorang penjenayah apabila melakukan jenayah dengan menggunakan komputer itu. Oleh kerana itu, objektif pertama projek ini adalah untuk mencari tempat di dalam rangkaian selain daripada komputer dan kemudiannya mengkaji anatomi sesebuah rangkaian itu bagi mempelajari kebaikan dan keburukan yang mana ianya berkaitan dengan keselamatan dan ketulusan sesebuah maklumat. Kunci penyelesaian bagi masalah ini adalah dengan menggunakan “Honeynets” yang mana ianya dapat memberi jaminan bahawa sesebuah maklumat digital itu memang boleh dipercayai. “Honeywall” pula adalah komponen yang paling penting dalam rekabentuk “Honeynet” yang mana ianya digunakan sebagai gerbang rangkaian yang tersembunyi. Walaubagaimanapun, “Bridging Mode” dalam sistem rangkaian adalah penting yang mana ianya tidak ditetapkan padanya Protokol Internet (Internet Protocol) dan bertujuan menjadikan sesebuah “Honeywall” itu halimunan. Ia juga berfungsi untuk melindunginya daripada dikesan dari dunia luar. Bagi mencapai tujuan projek ini, beberapa alat aplikasi telah dipasang di dalam “Honeywall”. Antara alat-alat aplikasi itu adalah “Snort”, “Sebek Sever/ Client Architecture”, dan “Log Server Architecture”. Aplikasi “Snort” digunakan untuk mengumpul dan memusat data daripada rangkaian kepada pengkalan data. Manakala “Sebek Sever/ Client Architecture” pula digunakan untuk merekod “key loggers”. Ianya dilakukan dibawah penyulitan protokol “encrypted protocol” seperti “Secure Shell (SSH)” dan kemudian memasukkan semua data-data yang direkod ke dalam pengkalan data. Fungsi “Log Server” pula adalah untuk merekod segala kegiatan yang berlaku pada pelayan “server” seperti status terkini pelayan, pemprosesan yang berlaku pada pelayan (daftar masuk dan keluar), kesilapan yang berlaku pada pelayan dan sebagainya. Objektif kedua bagi projek ini adalah untuk membuat perbandingan di

antara tiga jenis “Honeynets”. Perbandingan yang dijalankan berkisar mengenai keputusan-keputusan yang dihasilkan oleh setiap satu daripadanya. Ianya bertujuan untuk mengetahui dan menunjukkan jalan penyelesaian terbaik untuk isu yang berkaitan dengan keselamatan, masa dan kos bagi pemusatan dan pengumpulan bukti-bukti rangkaian. Objektif terakhir bagi projek ini adalah untuk menghasilkan Prosedur Operasi Piawaian (Guidelines).

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

Name		Description
ACK	-	Acknowledge TCP Flag
ARP	-	Address Resolution protocol
CIDR	-	Classless Inter-Domain Routing
DCON	-	Data Control
DMZ	-	Demilitarized Zone
DNS	-	Domain Name server
Gen II	-	Second Generation
GUI	-	Graphical User Interface
HIDS	-	Host Intrusion Detection System
HTTP	-	Hyper Text Protocol Protocol
IDS	-	Intrusion Detection System
IP	-	Internet Protocol
IPv6	-	Internet Protocol Version 6
ICMP	-	Internet Control Message Protocol
IPS	-	Intrusion Prevention System
IGRP	-	Interior Gateway Routing Protocol
LAN	-	Local Area Network
LKM	-	Loadable kernel Module
MAC	-	Media Access Control
NIC	-	Network Interface Card
NIDS	-	Network Intrusion Detection System
OS	-	Operating System
OSI	-	Open System Interconnection
RFC	-	Request For Comments
RIP	-	Routing information Protocol
SCP	-	Secure Copy
SMS	-	Short Message Service

SOP	-	Standard Operating System
SSH	-	Secure Shell
SSL	-	Secure Socket layer
SYN	-	Synchronize TCP Flag
SYN-ACK	-	Synchronize\ Acknowledge TCP Flag
TCP	-	Transmission Control Protocol
UDP	-	User Datagram Protocol
VSFTP	-	Very Secure File Transfer Protocol
VPN	-	Virtual Private Network

CHAPTER 1

INTRODUCTION

In this chapter, the project problem statement has stated and presented and then followed by project objectives in order to treat these problems. Then, other aspects such as project scope and aim have decided. Finally, the chapter concluded with project requirements that used to conduct project experiments.

1.1 Overview

Most organizations fight computer attacks using a mixture of various technologies such as firewalls and intrusion detection systems. Conceptually, those technologies address security from three perspectives; namely prevention, detection, and reaction. We, however, believe that a very important piece is missing from this model. Specifically, current technologies lack any investigative features. In the event of attacks, it is extremely hard to tie the ends and come up with a thorough analysis of how the attack happened and what the steps were. Serious attackers are skillful at covering their tracks. Firewall logs and intrusion detection alerts are unlikely to be adequate for a serious investigation. We believe the solution is in the realm of Network Forensics; a dedicated investigation technology that allows for the capture, recording and analysis of network packets and events for investigative purposes. It is the network equivalent of a video camera in a local convenience store (Almulhem A. and Traore I., 2006).

Network Forensics is an important extension to the model of network security where emphasis is traditionally put on prevention and to a lesser extent on detection. It focuses on the capture, recording, and analysis of network packets and events for investigative purposes. It is a young field for which very limited resources are available.

Marcus Ranum (1997); security expert, coined the term network forensics. He also introduced a network forensic system called Network Flight Recorder.

Network forensics is the capturing, recording, and analyzing of network packets and events for investigative purposes. When designing such a system, there are several challenges which include:

1) Data Capture:
    a) Where do the data should be captured?
    b) How much data should be captured?
    c) How do we ensure the integrity of the collected data?
2) Detection Efficiency: The system should detect attacks efficiently in order to trigger the forensics process. Therefore, it should accommodate for different detection approaches.
3) Data Analysis: After collecting the data, the system has to correlate them in order to reconstruct an attacker's actions.
4) Attacker Profiling: The system has to maintain information about the attacker himself. For instance, it can identify the attacker's operating system through passive OS fingerprinting.
5) Privacy: Depending on the application domain, privacy issues can be a major concern.
6) Data as Legal Evidences: For the collected data to qualify as evidences in a court of law, they have to be correctly collected and preserved in order to pass admissibility tests.

The Honeynet Project is a non-profit volunteer organization dedicated to computer security project and information sharing. The group developed the first operational Honeynet that is a network set up with intentional vulnerabilities; its purpose is to invite attack, so that an attacker's activities and methods can be studied and that information used to increase network security. Honeynet contains one or more Honeypots, which are computer systems on the Internet expressly set up to attract and "trap" people who attempt to penetrate other people's computer systems. Although the primary purpose of a Honeynet is to gather information about attackers' methods and motives, the decoy network can benefit its operator in other ways, for example by diverting attackers from a real network and its resources. The Honeynet Project, a non-profit project organization dedicated to computer security and information sharing, actively promotes the deployment of Honeynets (SereachSecurity, 2007).

In addition to the Honeypots, a Honeynet usually has real applications and services so that it seems like a normal network and a worthwhile target. However, because the Honeynet doesn't actually serve any authorized users, any attempt to contact the network from without is likely an illicit attempt to breach its security and any outbound activity is likely evidence that a system has been compromised. For this reason, the suspect information is much more apparent than it would be in an actual network, where it would have to be found amidst all the legitimate network data. Applications within a Honeynet are often given names such as "Finances" or "Human Services" to make them sound appealing to the attacker. A virtual Honeynet is one that, while appearing to be an entire network, resides on a single server. Since its formation in 1999, The Honeynet Project has grown to include 30 members of the security community from Canada, Israel, Netherlands, Germany, Australia, and United States (SereachSecurity, 2007).

“Centralized data collection in a corporate enterprise environment is important at many levels, one of the most important being that it allows security administrators and analyzers to monitor many systems in one central place. Once this logs information is has centralized, it also allows them to perform a more complete analysis and gives the ability to correlate events that have occurred throughout the enterprise. These centralized network’s logs can be used for networking investigation as alternative evidences. One of the problems that face networks is that having Honeynets located on different places and need to be able to share the information about attacks among other members. To conduct this function of valuable information, centralized data collection should be used (Jeff Dell, 2004)”.

1.2 Problem Background

Nowadays thinking of security implementation has become very necessary. It is used as primary component to protect systems and networks from attacks. There is variety of systems that used for these missions. Firewall and Intrusion Detection System (IDS) are famous examples. Within traditional network, they have integrated to protect it from attacks, unauthorized access or intrusion against the network. However, when a hacker tries to attack this network or any computer node that has incorporated within it, these protection system disallow the hacker to conduct his attack, or in the worse case when these system have not configured properly during attack it record or register all the activities of the network. These traces of network activities can be used later to reveal and discover the attack by network forensic analysts. Figure 1.1 depicts the structure of traditional network that is used widely nowadays.

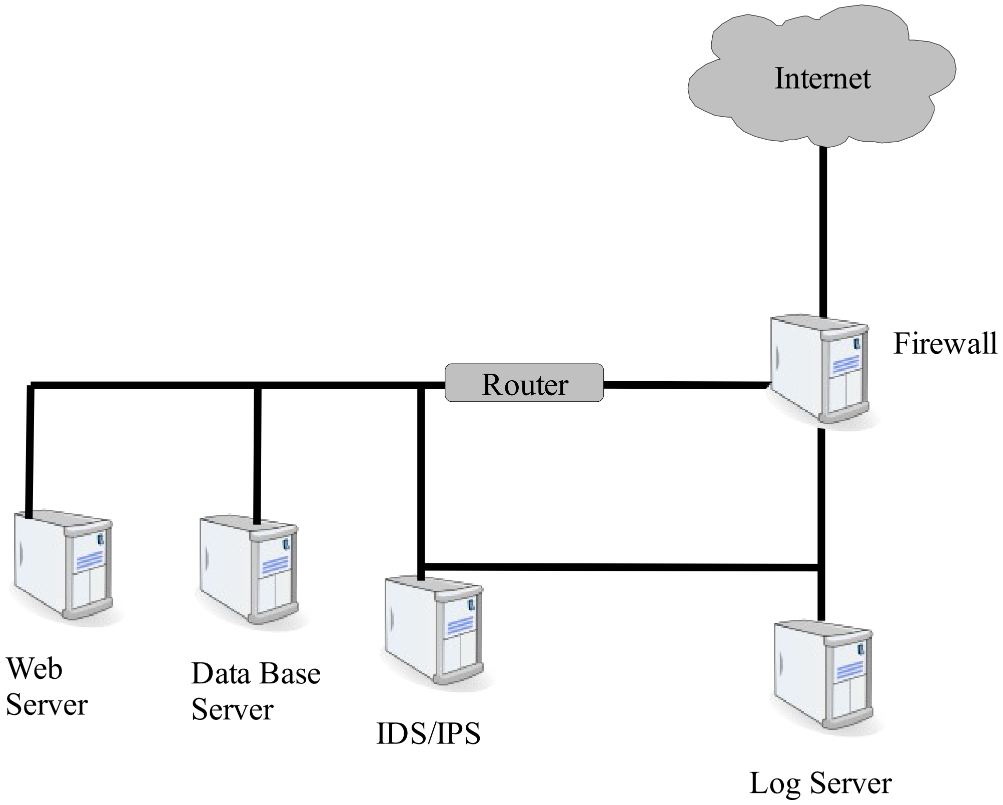

Figure 1.1 Traditional network.

Firewalls provide perimeter security and are used to protect trusted networks from untrusted networks. Not only they provide access control to the trusted network, but also provide access control from trusted networks to untrusted networks. The access control policy is defined with target network or machine address, target service and source network. Action , deny or permit , is attached to the policy. Whenever a new packet or connection is received by the firewall, the access control policy list is referred and action taken based on the action parameter defined in the matched policy (Intoto Inc, 2002).

The intrusion detection system in a similar way complements the firewall security. The firewall protects an organization from malicious attacks come through the Internet and it detects if someone tries to break in through the firewall or manages to break in the firewall security and tries to have access on any system in the trusted side and alerts the system administrator in case there is a breach in security**.** Moreover, Firewalls do a very good job of filtering incoming traffic from the Internet; however, there are ways to circumvent the firewall. For example, external users can connect to the Intranet by dialing in through a modem installed in the private network of the organization. This kind of access would not be seen by the firewall. Therefore, an Intrusion detection system (IDS) is a security system that monitors computer systems and network traffic and analyzes that traffic for possible hostile attacks originating from outside the organization and also for system misuse or attacks originating from inside the organization (SANS institute - IDS, 2001).

Recently, attackers have become more intelligent due to investigations. They have developed now tools that can be used to hide or delete the traces that might lead to catch them. For instance, like after they conducted an attack, they delete firewall logs files, or IDS logs files, or system logs files, and so on. Furthermore, sometimes they are using encrypted channel which make traces analysis impossible without decryption.

Therefore, this project cares with these problems and focuses on design and improve technologies that may lead avoid or to overcome them. Below are some quotes from forensic community show the importance of this project:

When encryption is not used, it is possible to monitor the keystrokes of an intruder by capturing the network activity off of the wire and then using a tool like ethereal to reassemble the TCP flow and examine the contents of the session. This technique yields not only what the intruder typed but also what the user saw as output. Steam reassembly techniques provide a nearly ideal method to capture the actions of an intruder when the session is not encrypted. When the session is encrypted, stream reassembly yields the encrypted contents of the session. To be of use these must be decrypted. This route has proven quite difficult for many. Rather than trying to break the encryption of a session, others have looked for a way to circumvent encryption (Honeynet Project, 2003).

Not everyone wants or needs to apply a Honeynet to achieve this function, but for those people who are intimately involved in security, they can be a valuable resource. A Honeynet allows an investigator or responder the ability to keep up with what attacks are occurring and how to recognize them and investigate them. Because of the flexibility needed, Unix and Linux systems make an ideal platform upon which to build a Honeynet. Almost each Honeynet's implementation relies on significant logging and control of network connections to be most effective, and the flexibility of networking options inside UNIX and Linux systems fit the bill perfectly (Daniel Hanson, 2004).

To learn about attack patterns and attacker behavior, the concept of electronic decoys or Honeypots are often used. This look like regular network resources (computers, routers, switches, etc.) that are deployed to be probed, attacked, and compromised. This electronic bait lures in attackers and helps with the assessment of vulnerabilities. As Honeypots are being deployed more and more often within computer networks, Blackhats have started to devise techniques to detect, circumvent, and disable the logging mechanisms used on honeypots (Thorsten Holz, Frederic Raynal 2005).

Information that is encrypted must at some point be decrypted for it to be of any use. The process of circumvention involves capturing the data post decryption. The idea is to let the standard mechanisms do the decryption work, and then gain access to this unprotected data. The first attempts to circumvent such encryption took the form of trojaned binaries. When an intruder broke into a Honeypot, he or she would then log into the compromised host using encrypted facilities such as SSH. As they typed on the command line, a trojaned shell binary would record their actions. To counter the threat posed by trojaned binaries, intruders started to install their own binaries. It became apparent that the most robust capture method involved accessing the data from within the Operating System's kernel. When capturing data from within the kernel, the intruder can use any binary they wish, and we are still able to record their actions. Further, because user space and kernel space are divided, there is ample opportunity to improve the subtlety of the technique (Honeynet Project, 2003).

Centralized data collection in a corporate enterprise environment is important at many levels, one of the most important being that it allows security administrators and analyzers to monitor many systems in one central place. Once this logs information is centralized, it also allows them to perform a more complete analysis and gives the ability to correlate events that have occurred throughout the enterprise. These centralized network's logs can be used for networking investigation as alternative evidences. One of the problems that face networks is that having Honeynets located on different places and need to be able to share the information about attacks among other members. To conduct this function of valuable information, centralized data collection should be used. Furthermore, guidelines which after testing become Standard Operating Procedures (SOP); that governs shape of network evidence extraction should be established.

“The rapidly evolving age of information network is changing our lives without our awareness. With the development of information communication technology (ICT) and cybercrime (Internet crime) intelligence, modern judicature that including criminal, civil and administrative must carry out litigation by using technology; especially in dealing with organized and terrible crime. Since digital evidence has often been very effective and important, the legislative and legal authorities in each country have gradually put more credence to digital evidence. As a result, establishing guidelines which after testing become Standard Operating Procedures (SOP); are important in raising the effectiveness and credibility of digital evidence. Subsequently, the move to create a digital evidence's standard operating procedure (DESOP) is essential to the development of a sophisticated information society. We would like to discuss the establishment of DESOP from procedure and software tool (Lin, A.C *et al*, 2005)”.

1.3 Problem Statement

The open issues described in the previous section lead to mentioning some project questions addressed in this project are follows:

a) How to design and improve a secure mechanism to collect network activities without attacker’s knowledge?
b) How to design and improve mechanisms to centralize the extracted network digital evidence?
c) What are the guidelines that govern and control data extraction and centralization operations?

1.4 Project Objectives

The main objectives of this project are how to improve collecting of computer digital evidences. Collecting of computer digital evidences will need to undertake a number of challenges. The following project objectives are in place:

i. To study concepts of network logs that can be used as evidences for network forensic and investigate challenges that they face.
ii. To design and improve robust alternative secure solutions for network that overcome network forensic challenges and can be used to extract and centralize network evidences without attackers' knowledge.
iii. To evaluate the usage of Honeynet types for monitoring and recording on line activities and demonstrate affects of their variables such as time, cost, and applied security in order to capture the network digital evidence.
iv. To create and establish guidelines those govern and determine how does network's evidence should be extracted and centralized.

1.5 Project Aim

The aim of this project is to design and improve mechanism to centralize the extracted network digital evidence which focuses on network logs that can be used by network forensic analysts to study attacks' analysis.

1.6 Project Scope

This project focuses on design and development of network evidence's centralization mechanisms for the evidence that has been extracted form network logs, since system forensic is beyond of the scope. The scope of this project covers the following points:

i. Reviews and comparisons of existing Honeynet types that will be used as secure solutions for network data collection.
ii. Design and improvement of technologies to be used for data extraction and

centralization from network logs.

iii. Creation and innovation guidelines which after testing become Standard Operating Procedures (SOP); those govern and decide how does network evidence should be extracted and centralized.

iv. Evaluation of the usage of Honeynet types for monitoring and recording on line activities and demonstration of their variables such as time, cost, and applied security in order to capture the network digital evidence.

1.7 Project Requirements

To effectively achieve the project's objectives that have been mentioned early, the following requirements and tools are needed:

i. Small network that shipped with several computers to be used to set up Honeynets.

ii. Special operating system that is called Honeywall to be used as a gateway of Honeynet systems for capturing and controlling inbound and outbound traffic through Honeynet systems. The following components of Honeywall also are important that used for setting up, configuring and using for collecting inbound and outbound data about attackers' activities. These components such as:

   a) Firewall: is called IPtables that used to control inbound and outbound of Honeynet systems;

   b) Snort_inline: is modified version from Snort software, that used for Intrusion Prevention System against attacks;

   c) Swatch software: it's used to notify administrators with attacks by send alarms, e_mail for example;

   d) Sebek server and client software: It's a hidden kernel module that used for capturing attackers' activities without their knowledge; and

   e) Encrypted channel to store attackers' activities remotely.

iii. Virtual Machine software: such as VMWare to test Virtual Honeynets.