POLICIES TO ENHANCE

THE FORENSIC OF COMPUTER SECURITY

Yanet Manzano

Date
April 19, 2000

Committee Member & Department:
Dr. Alec Yasinsac, Computer Science (Directing Professor & Department)
Dr. Yvo Desmedt, Computer Science
Dr. Ladislav Kohout, Computer Science
Dr. Stephanie A. Gore, College of Law

This thesis is dedicated to my parents, Peter and Candelaria Manzano. Thank you, for the midnight snacks those nights when I did not have time to go to sleep, the rides when I was to tire to drive, the words of encouragement when I got a low grade, and the unconditional support even whenever I started a new project. But above all else, thank you for loving me without conditions, without constrains. You are my light, my strength, my source of inspiration, and I thank GOD every day for making you my very own Guardian Angels. I dedicate this thesis to you, to recognize all the hard work you do. It never goes unnoticed.

- Yanet -

ACKNOWLEDGMENTS

Special Thanks, to Dr. Yasinsac. Thank you for doing much more than your job required you to do. Thank you to the members of my committee, Dr. Stephanie Gore, Dr. Yvo Desmedt, and Dr. Ladislav Kohout for taking time out of their busy schedules to participate in this project. Thanks to Justin, and the rest of my network Security Research Group. Special Thanks to Dr. Eva Asher, director of El Centro/ Hispano Latino Student Center, and to Dr. Patricia Stith director of the Retention Studies and Coordinator of Retention at Florida State University. Last but not least, thank you to the Honor Program at FSU, for giving me the chance to complete my project.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

DOS- Denial of Service Attack

IDS- Intrusion Detection System

IW- Warning and Indications

TCP - Transmission Control Protocol

IP - Internet Protocol

UDP - User Data Protocol

SMTP - Simple Mail Transfer Protocol

FTP - File Transfer Protocol

TFTP - Trivial File Transfer Protocols

AOL - American Online

OSI - Open System Interconnect

MIS - Management Information Systems

ISP - Internet Service Provider

INTRODUCTION

This paper is an honor's thesis, began from scratch last fall. My research last fall focused on learning as much as I could about computer hacking and the state of the art in computer forensics. This research gave me a broad perspective of computer crime and how they are investigated. After months of research, we decided that the goal of my thesis should be to propose policies that companies can establish to increase the chances of success of a forensic investigation if an attack occurs. So we came up with the title: Policies to Enhance the Forensic of Computer and Network Security.

The thesis is divided into nine parts. Part 1, is an introduction which will answer the traditional what, how, who, and when questions about the field of computer forensics. The rest of the thesis concerns the forensic model presented in Figure 1. This model describes the sequence of events that occur from the moment a computer attack starts, to the moment the attacker is discovered. The first event in the model is the attack. An attack is characterized by the series of steps taken by the attacker to penetrate the system, which we will discuss in Part 2, titled "The Anatomy of a Hacking Attack." The second event is the detection of the attack once the system has been penetrated. This is usually done by the Intrusion Detection System, which we will discuss in Part 3, titled "Taxonomy of Intrusion Detection Systems." The rest of the events in the model, which include emergency response, forensic investigation, collection of evidence, and attacker identification are discussed in Parts 4 to 9. These events are directly related to the field of computer forensics, and are discussed in the form of policies. They constitute the focus of the thesis, which is to establish policies that will enhance the computer forensics field.

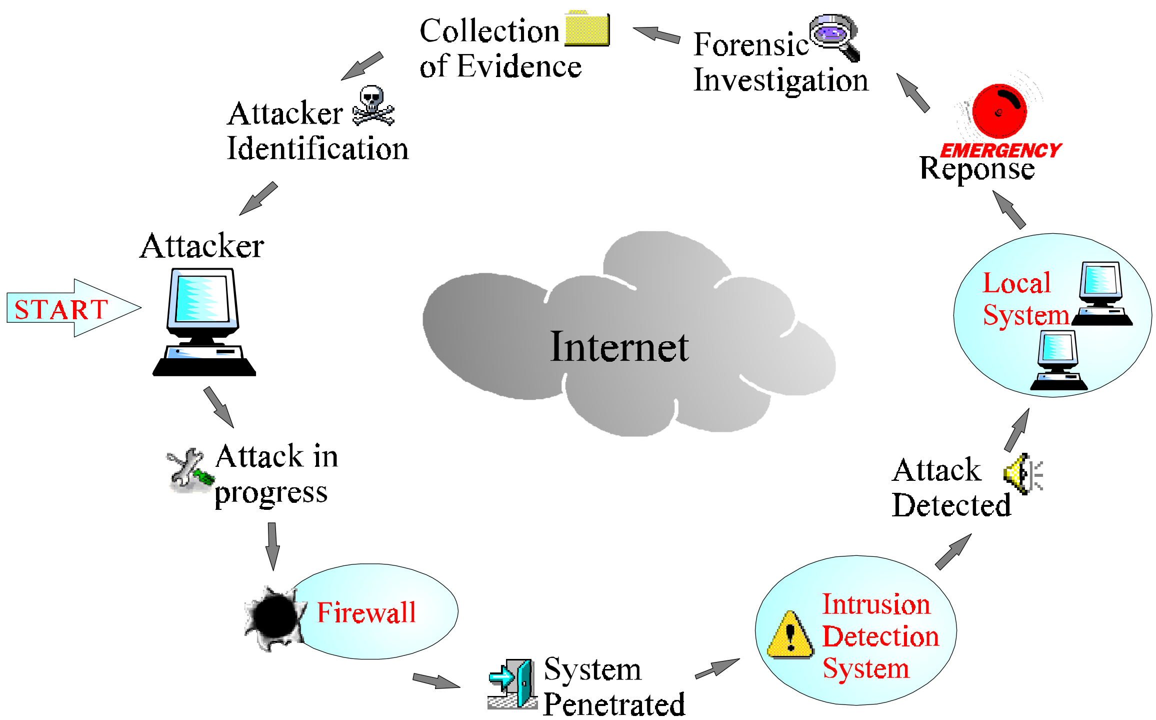

Figure 1: Forensic Model

PART 1

COMPUTER AND NETWORK FORENSICS

INTRODUCTION

"*Hackers got Bill Gates' credit card info. Self-styled 'Saint of E-commerce' arrested for e-biz hacks, e-mailed credit card details, including those of Gates, to NBCi* [RUTE00]." We see more headlines like that one in the news every day. They are evidence of the increase in the number of hacking attacks in what we call "the age of technology." As technology advances, computers become more powerful, with greater CPU speed and hard drive capacity, among other things. Unfortunately, these advances are not always put to good use, and although they bring great advantages, they also bring disadvantages. Computer crime is one of the disadvantages of computer use. Greater power increases the danger of abuse, and the security concerns. "Every system, whether linked to a wide area network via modem or not, is under threat. Security is not just a tedious shore anymore, but a necessity. But how can you protect yourself from this unseen intruder, without acting like an extra from a James Bond film? [EARL91]" The job is not easy, and preventing some crimes is not enough. It is also necessary to discover and punish those who bypass security systems and commit a crime.

As a response to these new security concerns, a new field in computers was created, the field of computer and network forensics. The field can be traced back to the 1980's, the first time that investigating authorities sought to present computer-based information as acceptable evidence in a court of law [ACF00]. Computer forensics is the art of discovering and retrieving information about a crime in such a way to make it admissible in court. In some cases, the field extends to data recovery. Experts in this field possess part of the answer to the current problem of computer crime. This part of the answer, as said in a computer forensics article, ". . . is thorough knowledge of the art of computer forensics. And that, in turn means 'knowing how' - partly like a lawyer, partly like a policeman, and partly . . . like a criminal [CF98]."

The question that comes to mind, then, is "How is this science used, and who uses it?" Computer Forensics is used to discover evidence in a variety of computer related crimes. The crimes can be theft of trade secrets, destruction of intellectual property, or misuse of computers, for example, an employee using company equipment to see pornography or to steal money. Most of the time, a breach in security is followed by an investigation to determine the cause, perpetrator, and damages caused by the crime. In order to assure the success of the investigation, it is necessary to follow a careful procedure. According to Judd Robbins, president of The National Forensic Center, and well-known computer specialist, this procedure is structured as follows:

1 Protect the subject computer system from alterations, damage, data corruption, or virus introduction.

2 Discover all existing, deleted, encrypted, hidden and password-protected files on the subject computer.

3 Recover all or as much as possible of deleted files**.**

4 Reveal the content of hidden files as well as temporary or swap files[1] used by application programs and the operating system.

5 Access the contents of protected or encrypted files**.**

6 Analyze all possibly relevant data.

7 Print out an overall analysis, including:

A A list of possible relevant files and discovered file data.

B Provide opinion of the system layout, the file structure discovered, any discovered data and authorship information, any attempts to hide, delete, protect, or encrypt information, and anything else that has been discovered and appears to be relevant to the overall computer system examination.

8 Provide expert consultation and/or testimony, as required.

Computer forensics experts are specially trained with the skills necessary to successfully carry out each step of the procedure mentioned above. Just like a detective in the police force, a forensics expert must have investigative skills, in order to supervise an investigation, and interview witnesses and suspects. Like a lawyer, a forensics expert must have legal skills to know which laws apply to computer crime as well as which laws

---

[1] Swap files: files used by the Operating System to do swapping. To swap means "to replace pages or segments of data in memory. Swapping is a useful technique that enables a computer to execute programs and manipulate data files larger than main memory. The operating system copies as much data as possible into main memory, and leaves the rest on the disk. When the operating system needs data from the disk, it exchanges a portion of data (called a page or segment) in main memory with a portion of data on the disk. [AOLC96]."

apply to evidence gathering. Like forensics experts that examine ballistic reports and perform autopsies, a computer forensics expert must have court room presentation skills since they may be called as an expert witness. Last, just like a criminal, a computer forensics expert must have the computing skills necessary to reconstruct the crime in order to acquire the necessary evidence.

The evidence found during a forensic investigation may depend on the type of crime committed. For example, in a criminal case, incriminating evidence may be found such as documents related to homicides, financial fraud, drug or embezzlement record-keeping, or child pornography. In a civil case, evidence of personal and business records related to fraud, divorce, discrimination, or harassment could be found. However, computer forensics experts are not only hired by lawyers, they can also be hired, for example, by an insurance company, to discover evidence that might decrease the amount to be paid in an insurance claim. Individuals may also hire computer forensics experts to support a possible claim of wrongful termination, sexual harassment, or discrimination. Nevertheless, computer forensics is not an exact science, so it can offer no guarantee that an expert will find any evidence, or that the data found will be useful at all.

Computer forensics is part of the answer to the computer crime problem, but can we leave it all to computer forensics experts? The answer is no. We can help by taking two steps. The first step deals with prevention. Computer forensics experts come into the scene after the crime is being committed, but the focus of computer security is to avoid getting to this point. Prevention is still the ultimate goal, so establishing policies and taking measures toward this end is the first step. The second step deals with measures to enhance computer forensics. These measures help increase the chance of success of a forensic investigation by establishing policies that increase the speed with which computer forensics experts collect evidence of computer crimes.

Now that we have a general idea of what computer forensics is all about, we are ready to proceed with Part 2, where we will analyze the attack. Part 2, includes a detail analysis of the anatomy of a hacking attack, and it relates to the first two events in the forensics model (Figure 1), “attack in progress,” and “system penetrated.”

PART 2

ANATOMY OF A HACKING ATTACK

Enud Tenebaum, David L. Smith, and Kevin Mitnick are three names that are very well known in the hacking world. Enud Tenebaum, alias "The Analyzer," orchestrated what was called the biggest break-in in the Pentagon. He also claims to have hacked computers at Harvard, Yale, MIT, NASA, and The US Navy. David L. Smith was the creator of the Melissa virus which caused 80 million dollars in damage. Kevin Mitnick, alias "The Condor" and the most famous hacker in the world, stole codes from "tech" companies including Sun Microsystems. The question is, "How did they do it?" We answer that question by analyzing the anatomy of a hacking attack.

One of the most common types of attacks is the Denial of Service Attack (DOS). DOS attacks may be aimed at devices and networks with exposure to the Internet. The goal of this type of attack is to disrupt or deny service to legitimate users, networks, systems, or other resources. The technique used by this type of attack is to trigger a program that will generate enough traffic to a site that it denies service to the site's legitimate users. The SYN attack and the Teardrop attack are two of the most commonly known types of DOS attacks.

The SYN attack is executed during the three-way handshake that kicks off the conversation between two applications. The three-way handshake connection goes as follows: Step 1) A client sends a request for a connection to a server, Step 2) The server answers the client with a message indicating is ready and waiting to make the connection, Step 3) The client sends a confirmation, and the server goes on to establish the connection. During a SYN attack (Figure 2), the attacker completes Step 1 several times posing as legitimate users. The server in turn completes Step 2 and sends the messages to the users apparently asking for a connection. Since these users never requested a connection, they will never complete Step 3 and the server will be left waiting for confirmation. Parallel to this process, legitimate users trying to connect to the server are denied a connection because the server is busy answering all the requests made by the attacker.

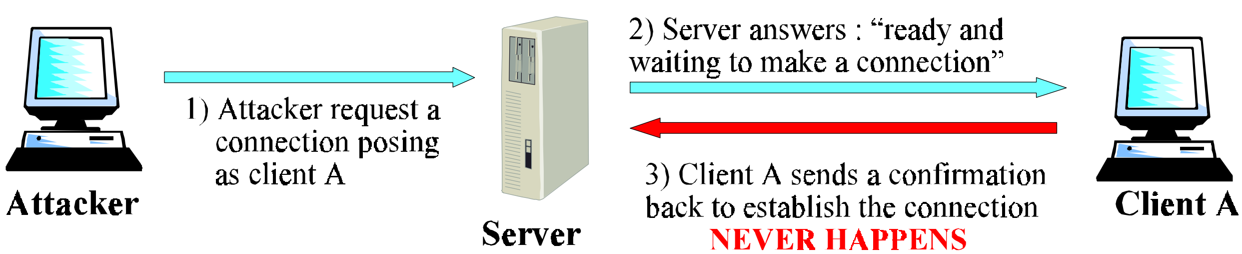

Figure 2: Schematic of the SYN attack.

The Teardrop attack, on the other hand, occurs during the transmission of a packet. When a message is sent through the internet, it is broken down into packets, and each packet may eventually be broken up into smaller chunks (Figure 3). Each fragment looks like the original IP packet except it contains an offset field that says, for instance, “This fragment is carrying bytes 600 though 800 of the original (non-fragmented) IP packet.” The Teardrop program creates a series of IP fragments designed to overload offset fields. When these fragments are reassembled at the destination host, some systems will crash, hang, or reboot [DOWN98].

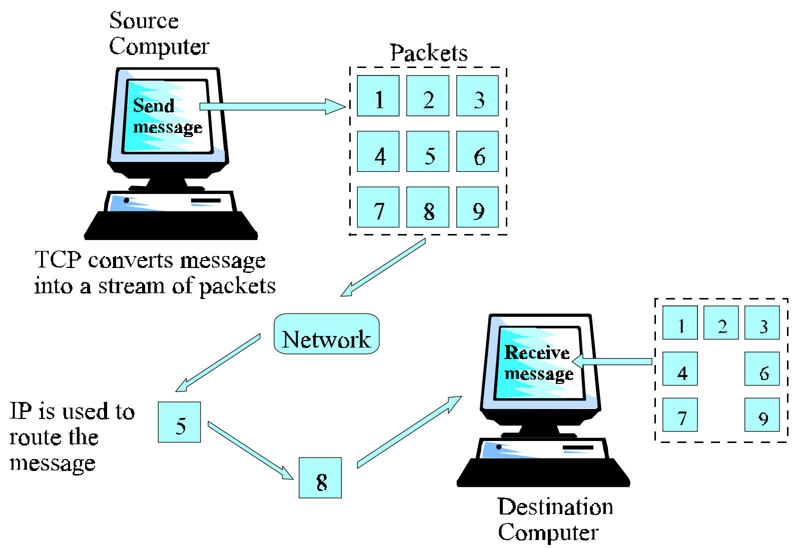

Figure 3: Schematic of a message split into packets and sent over the Internet [GREE99].

DOS attacks are becoming very common because they require few skills, and they do not require any break-in or theft of sensitive information in order to be successful. However, we are more concern with the type of attack that does requires a certain level of expertise by the attacker. Each attack is unique, but to be able to analyze them we can follow a general model. This model includes six steps: footprinting, scanning, enumeration, penetration, advance, and covering tracks (Figure 4, next page). Not all attacks follow this model step by step. Some attacks will combine two of these steps, or

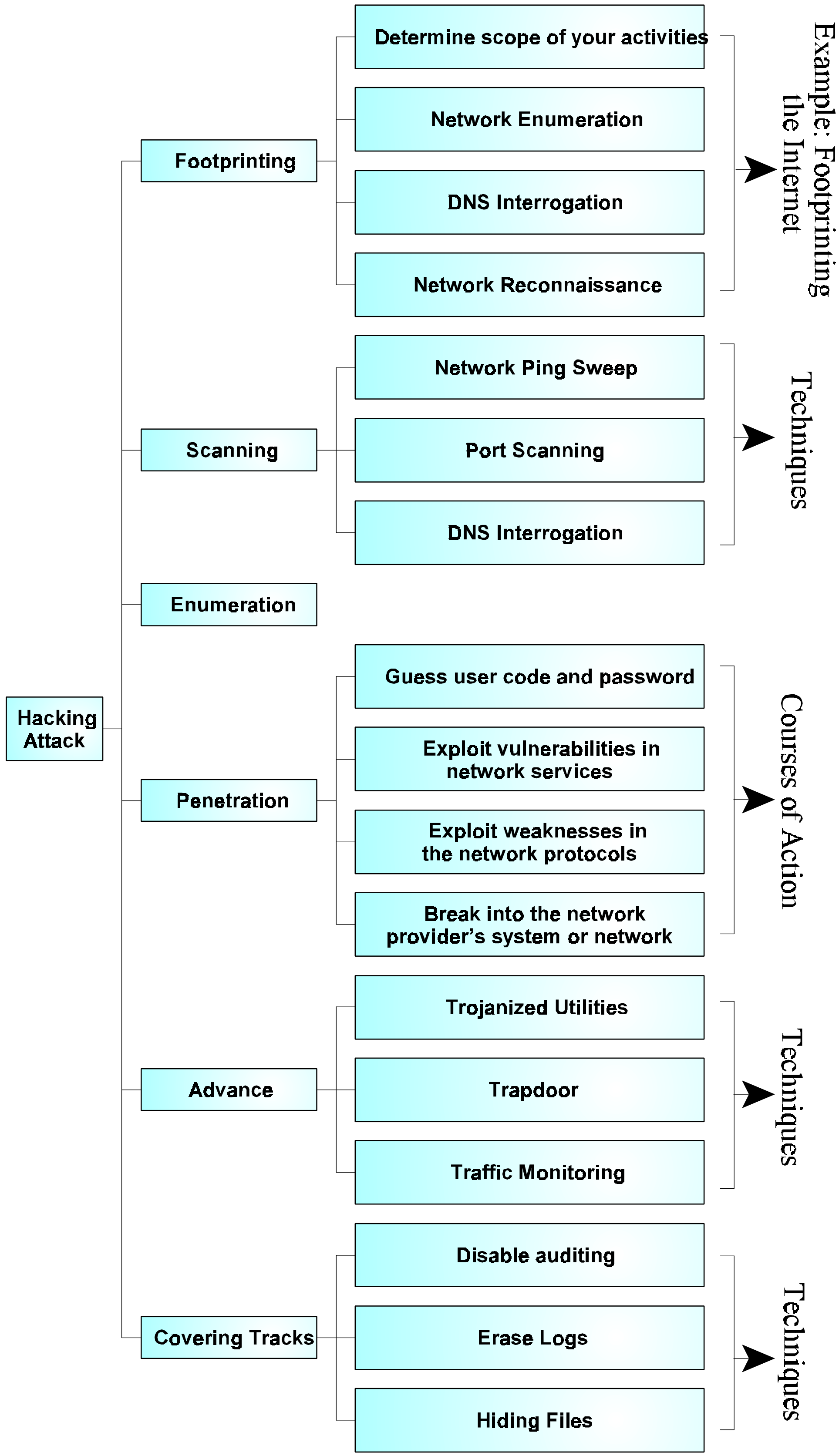

Figure 4: Anatomy of a Hacking Attack.

skip a step.

Footprinting is the first step in the anatomy of a hacking attack [MCCL99]. The objective of this step is gather information that is essential to the attack, such as the address range and naming acquisition. Every piece of information is useful, no matter how insignificant it may seem. This step is also known as intelligence gathering, and it allows attackers to create a complete profile of an organization's security posture.

There are many types of footprinting techniques. They are primarily aimed at discovering information related to Internet, intranet, remote access services, and extranet. These techniques include a series of steps similar to those mentioned in the following example: footprinting a company's internet connection [MCCL99] (Figure 5).

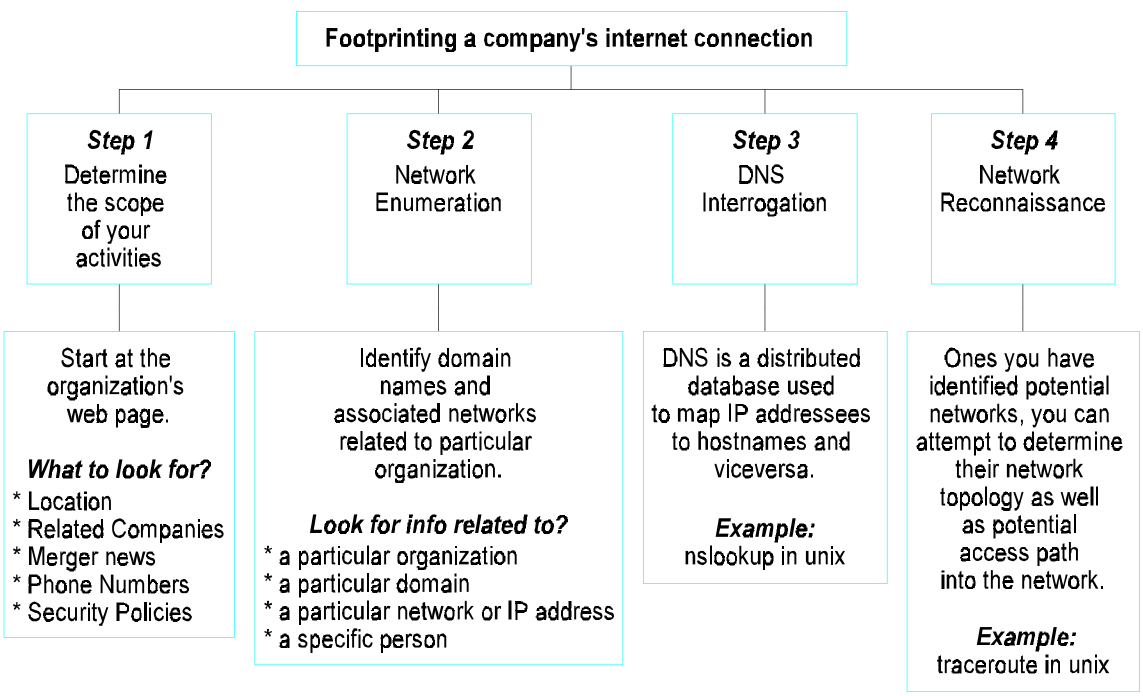

Figure 5: Footprinting Example

Scanning is the second step in the attack. While footprinting provides a list of network and IP addresses, scanning determines which systems are live and reachable from the Internet. Among the different techniques used in scanning are network ping sweeps, port scanning, and operating system detection. Network ping sweeps are one of the basic steps in mapping out a network. A ping sweep is performed on a range of IP addresses and network blocks to determine if individual network systems are alive. Port scanning is designed to search a specific system for transmission control protocol (TCP) and user data diagram protocol (UDP) ports offering services. The port service will attempt to connect to each port in turn, verify whether the connection is accepted, and identify responses sent by utilities such as Telnet and Sendmail [simple mail transfer protocol (SMTP) electronic mail] to check for known vulnerabilities. The operating system detection technique identifies the type of operating system of the target system to determine vulnerabilities [MCCL99].

Enumeration is the third step in the attack. Enumeration seeks to identify active connections and directed queries. The attacker's probing process starts here since he may have acquired valid user accounts or poorly protected resource shares. In this step the attacker will use different techniques depending on the system, whether Windows, Novell NetWare, Unix or some other system. An example of a risky command common to many systems is "finger." Finger helps the attacker determine active users in the system, peak usage time, and periods when no system administrator is logged on. Enumeration is fairly simple because it is largely a continuation of scanning, and very important because it provides the information needed for penetration. However, it is also dangerous because frequently, activities from this point are recorded in log files [MCCL99].

Penetration is the fourth step in the attack. After enumeration, enough data have been gathered to make an informed attempt to access the target exploiting security weaknesses [MCCL99]. There are different levels of penetration. For example, if the system has obvious configuration errors, such as access via FTP or trivial file transfer protocols (TFTP) to the full file system, then access is gained quickly. Otherwise, according to David Ferbrache, and Stuart Mort in their article, "Malicious Software and Hacking," the intruder has four courses of action [FERB97]:

1. Try to exploit vulnerabilities in network services, such as Sendmail, to gain access to the system, for which the attacker can use a variety of scripts that are freely available.

2. Try to guess user code or password. This can be accomplished automatically with a password cracking software. Some types of cracking software use a dictionary approach, by entering each word in various dictionaries as passwords to see if a match is found.

3. Try to exploit weaknesses in the network protocols, such as those that allow IP spoofing[2].

4. Try to break into the network provider's system or network to capture

[2] IP spoofing: "A technique used to gain unauthorized access to computers, whereby the intruder sends messages to a computer with an IP address indicating that the message is coming from a trusted port. To engage in IP spoofing, a hacker must first use a variety of techniques to find an IP address of a trusted port and then modify the packet headers so that it appears that the packets are coming from that port [AOLC96]."

user codes and passwords passed in the clear[3] across the provider's network.

Advance is the fifth step in the attack. At this point the attacker has successfully penetrated the system, so the next logical step is to expand. The objective of this step is to exploit system configuration errors and vulnerabilities to gain additional privileges, such as breaking into root accounts[4]**.** For example, the attacker may install Trojanized[5] utilities, such as Telnet, to record passwords used to access remote systems. The attacker may also install a trapdoor to allow easy access for hackers with knowledge of a special password, or to monitor traffic on the local area network to gather password and access credentials passed in the clear.

Last, but not least, the attacker turns to the sixth step, covering tracks. Once the goals of the attacker are accomplished, hiding this fact from the system administrator and users among others becomes the priority. Usually, the first thing an attacker will do to cover his tracks is to install a rootkit[6]. A rootkit allows an attacker to disable auditing, clear event logs, and hide files [MCCL99]. Normally, systems keep the auditing option turned on to keep records of events that might have security significance, such as when

---

3 "passed in the clear" means that the data sent across the network is plaintext.

4 Root account: account with privileges similar to those of a system administrator, which allows complete access to the system.

5 Trojanized utility: a program that appears to do something, perhaps because of their name, such as DIR or Telnet, but when is executed it does something completely unexpected, such as format the hard drive.

6 Rootkit: a group of computer tools, or softwares.

access to resources occur. In order to avoid detection, it is imperative for an attacker to avoid recording of his actions, so he needs to disable auditing tools. In the event that actions were recorded, it is necessary to clear the event log by deleting the system files that record information about what is being done on the system, like copying or deleting files. One additional task also performed by the rootkit is the hiding of any files the attacker left behind, such as trojanized programs, or files to install a trapdoor.

Now that we have a general idea of how an attack is done, we are ready to proceed with Part 3, where we will analyze how these attacks are detected by a computer system. Part 3, includes a detail analysis of the taxonomy of Intrusion Detection Systems, and it relates to the next two events in the forensics model (Figure 1), “attack detected,” and “emergency response.”

PART 3

TAXONOMY OF INTRUSION DETECTION SYSTEMS

We never know when a hacking attack will occur. It could be 2 in the afternoon, or 2:30 in the morning. It is impossible for one person to watch the network 24 hours a day, and even if there was someone to do it there is no guarantee that a person will detect every attack. So, how do we fight against this constant threat? One way to fight back is to use Intrusion Detection Systems (IDS). An IDS "monitors the actions taken in a given environment, and decides whether these actions are symptomatic of an attack or constitute a legitimate use of the environment [DEVA99]." If the actions are determined to be symptomatic of an attack against a computer system[7], or network, or against an information system in general, then it signals that an attack is in progress, and reacts according to the countermeasure it is programmed with.

IDS analysis is based on four characteristics which fall under two types, functional and nonfunctional characteristics (See Figure 6). The first three characteristics are detection method, behavior on detection, and audit source location;

---

7 "System is used here to denote an information system being monitored by an IDS. It can be a workstation, a network element, a server, a main-frame, a firewall, a web server, an enterprise network, etc [DEVA99]."

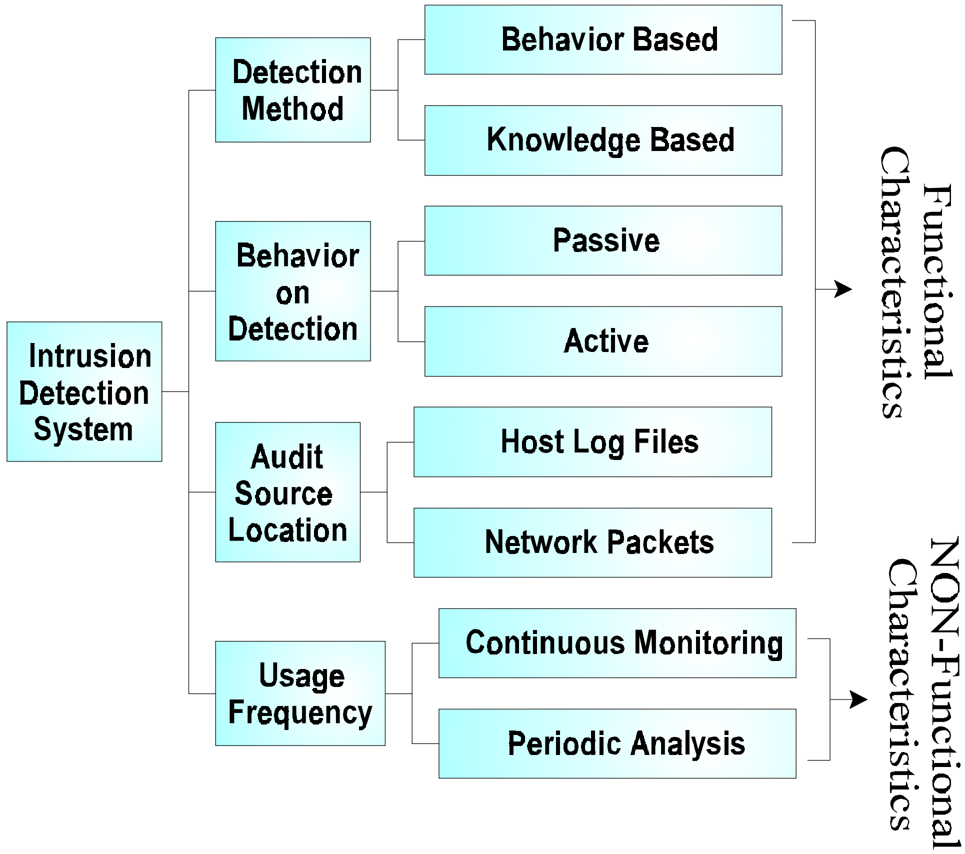

Figure 6: Characteristics of Intrusion Detection Systems [DEVA99].

they fall under the functional classification because they have to do with the internal functioning of the system. The last characteristic is usage frequency, which falls under the nonfunctional classification. It is necessary to consider all of these factors before choosing an IDS.

There are two types of detection methods currently in use, knowledge based and behavioral based. Knowledge based systems work in a way similar to virus scanners. They collect information about attacks and system vulnerabilities, and create a signature for each attack. If the IDS detects a match for an attack signature, which can be a sequence of commands, then it triggers an alarm much in the same way a virus scanner prompts the user when a virus is found. The disadvantage with this system is that much like virus scanners, if a new attack is used then it will go undetected since the system cannot match it with any of the known signatures.

Behavioral based systems, on the other hand, assume that an attack can be detected by observing a deviation from users normal behavior [DEVA99]. These systems build user profiles instead of attack signatures. It establishes parameters around which these profiles should vary. If the deviation of a user profile is greater than that established as normal, then an alarm is activated to signal a possible attack. This system, however, also has problems because it relies on user profiles. It does not account for new users, since there is no profile to compare activities against. It is also very resource intensive since the system must constantly update the profile of the users. Furthermore, the system can be defeated by users if they know it is being used, since the user may change its activities within the parameters, and change them slowly until he has gotten to the point where his attack will look to the system as acceptable behavior.

The second characteristic to consider when analyzing an intrusion detection system is behavior on detection. This characteristic has to do with the way the system behaves when an attack is detected, which could be passive or active [DEVA99]. Most systems are passive, which means they set off an alarm when an attack is detected, but no countermeasure is taken. Active systems on the other hand, set off an alarm and then go on to apply countermeasures to stop the attack, or to correct the discovered vulnerabilities.

The third characteristic is the audit source location, which is based on the kind of information the IDS analyzes, whether it is audit trails, system logs or network packets [DEVA99]. For example, a host-based IDS bases its findings on information gathered about an individual on a machine. The problem with this method is that it does not deal with network packets. Today many companies depend on the Internet. For example, consider, an online trading site. If they suffer a DOS attack, like a SYN attack, their customer will not be able to access their accounts and the company will lose money. A host-based IDS will never detect an attack like this one, since it does not analyze network packets, so for this company it is better to use a network-based information source.

The fourth characteristic has to do with whether the monitoring is done continuously or periodically. Continuous monitoring will be done by a dynamic IDS which performs real time analysis by acquiring information about the actions taken on the environment immediately after they happen. A periodic analysis, on the other hand, is static. It takes a snapshot of the environment and analyzes this snapshot, looking for vulnerable software, configuration errors, and so on [DEVA99]. Continuous monitoring would be the ideal solution, since the system would catch a problem right away. The problem with this method is that it consumes a lot of resources since the IDS has to analyze, and decide whether to react or not to every action that occurs on the system. Periodic monitoring uses a lot less resource, but it runs the risk that an attack was done during the period that the system was not monitored. It may even give a skillful attacker enough time to cover his track, so that the system may not be able to detect the intrusion.

Now that we have a general idea of how systems detect attacks through the use of

IDSs, we are ready to proceed with Part 4. Part 4 is an introduction to the rest of the thesis. The last five parts of this thesis deal with the last three events in the forensic model (Figure 1), “forensic investigation,” “collection of evidence,” and “attacker identification.” This last three events are what constituted the computer forensics process, and our goal from now on is to propose policies that will help enhance this process.

PART 4

POLICIES: INTRODUCTION

As stated above, with the rapid grow of the Internet and technology, the field of computer science has advanced greatly. This progress has brought great advantages for society, in fields like health, and education. Unfortunately, it has also brought a problem with computer crime, since the attacks get more sophisticated. As discussed in Part 2, a hacking attack is not just sitting in front of a computer one day and breaking in anymore. Hackers have a well thought out, and detailed plan to ensure success.

Preventive measures are not enough anymore. It is also necessary to have the ability to catch and prosecute attackers on the network, and this is, as we saw in Part 1, what the computer forensics field is all about. Since network owners cannot prevent crime, the following parts propose policies that will help decrease crime by enhancing the ability to catch and prosecute attackers.

PART 5

POLICIES: GATHERING POTENTIAL EVIDENCE

5.1 Introduction

Computer forensics, as discussed in Part 1, is the art of discovering and retrieving information about a crime in such a way that the information is admissible in court. The evidence is the cornerstone of this field. But what exactly is the evidence? The evidence is the information collected by different components of the system, such as the IDS, that might contain information about the attack. The information collected by the system does not become evidence until a crime is committed and this data is used to find clues. So, we call the data collected by the system potential evidence. But what kind of potential evidence would be useful in this field?

Logging is a vital source of potential evidence to the field of computer forensics. It allows us to go back and look at what happened in the past. The type of information kept in logs will depend mostly on the configuration of the system. For example, suppose we have a small company that is connected to the Internet through an online service, such as AOL or CompuServe. Then the company may not be able to keep logs of network router activity, since these devices are managed by the online company. In any case, we can still establish guidelines about the kind of potential evidence that should be logged, where it exists.

5.2 Files

Files are the most obvious source of evidence during an investigation, especially encrypted, deleted, and hidden files. In fact, Steps 3, 4, and 5 of the procedure followed by a computer forensics expert mentioned in Part 1 deal with the discovery of this type of files. These types of files often contain valuable evidence, but accessing them can cause legal problems. Revealing the content of these files has legal implications since it may be considered an invasion of privacy. It is necessary for a company to establish a policy that explains that employees have no expectation of privacy whatsoever, and that the company has the right to access any file in its system without permission, no matter who the file belongs to. Otherwise, if evidence is found in the files, and the case goes to court, the employee may be able to claim to have had a "reasonable expectation of privacy," regarding the files. If it is proven in court that the employee did not know the company could access his files without asking, then the court may decide that the evidence was gathered illegally because it violated the employee's privacy. In this case, the evidence gathered may not be admissible. Worse yet the employee may turn the situation around, and sue the company for invasion of privacy.

5.3 Network Devices

Network devices such as routers, and servers are a very good source when it comes to collecting Internet related evidence. A router is "a special purpose computer that directs data packets along the network [GREE99]." A server, on the other hand, is part of the client-server model. Servers "are computers that answer requests for services, such as list servers, mail servers, and news servers [GREE99]." It is necessary for companies that use these network devices to keep logs of the data packets that flow through them. Keeping records of these data packets, allow us to monitor the network in a static fashion. One of the key features of a packet is that it contains an origin and destination address in addition to the data. These features are of particular interest during a forensic investigation. The origin address may provide us with information about the attacker's identity, the destination address may provide us with information about a specific target within a system, and the data may provide us with information about the attack itself.

Attackers use many different techniques to hide information in a packet. Some of these techniques are based on Steganographic methods. Steganography "literally means, 'covered writing' and encompasses methods of transmitting secret messages through innocuous cover carriers in such a manner that the existence of the embedded messages is undetectable [JOHN98]." The information may be hidden in text, or embedded in an image. However, because of a vulnerability in the protocols in the OSI model network, an attacker can also use a TCP/IP packet to send hidden information through the Internet [JOHN98].

In Part 2 we discussed the SYN attack. This discussion explained how when a message is sent through the Internet, it is splits into TCP/IP packets, then reassembled once all the packets are received (Figure 2). Each of those packets contains a header which includes information about the source and target computer, a sequence number to be able to reassemble all the packets, and error checking information. Each TCP/IP packet header contains, in addition to the space needed to store all that information, six extra bits that are left empty when the packet is sent. Most firewalls are configured to check these six bits, and thrown away packets that have information in them. However, the information in these six bits may be relevant if an attack occurs. So, if a system finds one of these packets, it should redirect it to a safe place and keep a record of it.

In addition, a system should also keep records of network events such as logging in, or out of, a computer and accessing network services such as remote Telnet, or FTP sessions. These record are very useful during an investigation because they contain information about the activities of a specific user, as well as dates, and times of those activities. This information correlated with internal events such as email or web access, and external events such as phone records, witness testimony, and physical evidence, can provide a “Time-Line” [HOSM98]. The Time-Line serves as a guide to place the different events in the system, and correlate them to the whereabouts of a suspect, helping either to establish an alibi or to discover incriminating evidence.

5.4 Intrusion Detection Systems Output

It is important to keep all the IDS output even if no attack was detected. The output from IDS is of particular importance to the computer forensics field, since these systems are in charge of analyzing data and signaling if evidence of a possible attack is found. They can be considered a computer forensics tool, since the data they analyze to determine if an attack occurred is evidence of the attack itself. However, as we saw in Part 3, IDS are not 100% effective. Depending on the different characteristics of the IDS in use, it is possible for an attack to go undetected. For example, we saw that a knowledge-based IDS works by comparing data to known attack profiles, but an attack that is not in the system's database will pass undetected. However, this is not to say that the information collected by the IDS is useless. Even if an attack was not detected right away, the output of the IDS can still be analyzed later on by a forensics investigator, and used to prove an attack actually occurred.

5.5 Email, and Web Access Information

Email, and Web access information should be kept, since they can reveal incriminating evidence upon inspection. If an attack occurs, and an employee is suspected of being the attacker or an accomplice, it may be necessary for the forensic investigation to check the employee's email and web access information for traces of incriminating evidence. It may even be necessary to monitor employees' activities for a period of time to be able to gather evidence. In both cases it is vital that we don't alert the employee of the investigation against him, otherwise evidence may disappear. In this situation we face a problem similar to that explain in section 5.2 when talking about files as potential evidence. If an employee does not know that the company can monitor his activities, and the company does it, then even though he was using company equipment the court may find he had a "reasonable expectation of privacy." If that happens then the evidence gathered may not be admissible, and the employee has the right to turn back and sue the company for invasion of privacy [CF98]. The solution to this problem is similar to that explain in section 5.2. It consists in establishing a policy that says what is and what is not acceptable use of company equipment, and that an employee has no expectation of privacy whatsoever when using company equipment [CF98].

5.6 System Documentation

Printers, tape drives, optical drives, hardware manuals and software manuals should be kept, since they may be needed to examine the potential evidence. Technology advances very fast. Both hardware and software are replaced very quickly, sometimes by entirely new programs, and sometimes by newer versions. Because of this constant change, it is very difficult to keep up with the documentation of all the software and hardware in use. It is very easy to throw out or lose outdated material, especially when a lot of new material is coming in to replace it. However, the ability to process and examine the potential evidence may be directly tied to special hardware, software and/or written instructions contained in manuals. Computer technology changes very quickly, and it may be impossible to obtain similar or outdated hardware or instruction manuals from other sources.

PART 6

POLICIES: PLANNING A RESPONSE

6.1 Introduction

Once we have collected the potential evidence, we have to make sure it is not destroyed or damaged by the people that come into contact with it. We would have no problem if we knew that the evidence will only be handled by experts, but in today's society most people that work with computers are users, not experts. Ignorance is a sure path to destruction, but then so is knowledge without a plan. The first step to ensuring the safety of the evidence is planning a response.

6.2 Setting Up a Response Team

A quick and effective answer to an attack may be the key to preserve the evidence of the crime, and the first step to implement an effective response is to create a response team. A team that is trained to handle the situation in the best possible manner [FAQN00]. According to Robert Graham, a well-known security expert currently working with Network ICE Corporation, a response team should include:

- Upper Management
- Human Resources
- Technical Staff

- Outside Members
- Security Team

First, we need to identify someone from upper management with the authority to handle delicate issues. Second, there should be someone from the department of human resources, provided one exists. They will be the people in charge of dealing with inside attackers, whether they are breaking into a machine, or browsing inappropriate servers looking for files like customer lists that might be left open. Third, there should be technical staff. Security is often handled separately from normal MIS activity. If security personnel detects a compromised system, they need to know who in MIS needs to be notified. Fourth, the team should also include people outside the company, who may need to be contacted in case of an emergency, such as an ISP person, or the police. Last but not least, we need to include a security team.

The actual structure of the security team will depend on the needs of the company. Looking at the example used in Part 5.3, where we had a company that is connected to the Internet through an online service such as AOL, or CompuServe, we can see that a response team may not be necessary. In this case, handling an attack may be entirely up to the online service, since they are the ones managing the network used by the company.

6.3 Setting Up a Response Procedure

A response procedure is a simple and prudent step by step guide that employees can follow if an attack is suspected. A procedure may include the following principals:

1. Don’t panic

2. Don’t alert the suspect

3. Don’t compromise the evidence

4. Alert the response team

One of the most common mistakes when an employee discovers a fraud or an attack is to panic. Panic may cause that employee to go after the attacker, destroying evidence in the process, or to simply ignore the attack for fear of being suspected or fired. The second most common mistake is to unintentionally alert the suspect. The employee must tell as few people as possible about his suspicion, since the attacker may be siting in the next cubicle. The third most common mistake is to compromise the evidence. The safest course of action in the case of an attack is to assume every computer has been programmed to destroy evidence [CEP99]. It is relatively easy for criminals to rig their computers to destroy hard disk drive’s content or specific files by planting decoy programs or through the modification of the operating system [EF99]. So, it is better to avoid running any programs, or commands, since the intruder may have installed trojanized utilities, and even a simple DIR command to view the content of a directory can destroy the content of the entire computer. The last, step is to alert the response team.

6.4 Setting Up an Investigative Procedure

The fact that a company is alerted about a possible attack, either by the IDS, or by an employee, is no guarantee that an attack has actually occurred. Employees make mistake, and IDS system are not 100% effective. However, since we do not know, we should always assume the worse and react as if an attack has actually occurred. Once the response team is alerted, and immediate action has been taken, the next step is to start an investigation. At this point it may not be clear whether a computer forensics expert is needed or not, so the company will most likely need to start a preliminary investigation. This investigation may be carried out by the security team, or technical staff part of the response team, or by an entirely different team. In either case, the investigator must examine the potential evidence to find out what happened. This potential evidence is the same that a forensics expert will need to use to prove the attack in court provided the company decides to go to court. Therefore, the preliminary investigation should be carefully conducted not to damage or corrupt the evidence. If the investigation is done in the right way, then the job of the computer forensics expert will be much easier because part of the evidence will have already been collected.

The procedure to follow during a preliminary investigation is similar to that mentioned in Part 1, and followed by a computer forensics expert during a forensic investigation. However, since the preliminary investigation is not as rigorous as the investigation carried out by a computer forensics expert, the procedure for it is also less rigorous. Here are the steps:

1. Make an exact copy of the machine that is being compromised using a disk imaging tool, and work on that copy so that the evidence remains intact.

2. Determine the exact nature of the computer crime or abuse and whether is ongoing or complete.

3. Identify how the crime was committed and hardware, and software involved.

4. Determine the sources of evidence, such as deleted, encrypted, protected files

5. Determine whether the crime was a solo effort or relied upon collusion and whether the perpetrator(s) came from within or from outside the organization

PART 7

POLICIES: TRAINING PERSONNEL

7.1 Introduction

Now that we have the plan, we have completed the first step to protect the potential evidence. The next step is to train the personnel with the skills necessary to implement the plan. In Part 6, we set up a response team; now, we have to prepare the team members for the job they have to do. We also need to train the rest of the personnel, to make sure they know the procedures that are in place and how to use them.

7.2 Training the Response Team

Once a response team is assembled, the members of the team need to be prepared for the kinds of decisions they will have to make. For example, one of the more difficult questions to deal with during an attack, says Michael Anderson, a former IRS high-tech investigator who founded New Technologies Inc. in 1997, is "To pull the plug or not? Is it nobler to lose all temporary files by cutting off power without a proper shutdown? This is a problem if criminals use an uninterruptible power supply and do all their work in memory. Or is it nobler to shutdown properly and risks a booby trap that wipes out data at the touch of a key? [RADC98]" Most of the time, this is a no-win situation. The best we can do is calculate which will cause the minimum losses. For example, if the company under attack is an online trading service, pulling the plug might be the best answer. Going off-line on such a service will have a major impact -- but would still be better than hackers trading away people's stocks.

7.3 Training The Investigative Team

The investigative procedure that follows an attack needs to be carried out with certain precaution, so the investigative team must have computer forensics skills. We have to make sure the investigative team members have the abilities necessary to follow the investigative procedure we mentioned in Part 6.4. This team does not need to have the full range of forensics skills that experts have, as mentioned in Part 1, because it is mostly a preliminary investigation. However, they do need to have at least investigative, and computing skills.

During a preliminary investigation, the investigative team will use these skills to determine whether an attack actually occurred, and if possible to identify the crime by determining how it was committed and who did it, and find the evidence left behind. In order to do this, the investigative team needs to understand the steps followed by the attacker, so that they can be traced back. In other words, the investigative team must know the anatomy of the attack, as we mentioned in Part 1. The team must also know where to find possible evidence. One of the first places to look is the output of the IDS. In order for this information to be useful, the investigative team must know the taxonomy of an IDS, as explained in Part 2, and understand the characteristics of the IDS in question. They must also use the evidence found to determine who committed the crime, if it was a solo effort or a collaborative effort, and whether the attackers came from outside or inside the company.

At this point, the investigative team has enough information to decide on a course of action, whether it is to call a computer forensics expert to gather evidence legally, and prosecute the suspect, or to fix the vulnerabilities that allow the perpetrator to enter the system, and leave it at that. If the decision is to prosecute the attackers, then it will be of great help if the investigative team has the full range of forensics skill, since then there would be no necessity to hire an outside expert.

7.4 Training General Personnel

It is necessary to train all personnel in company security policies. The general personnel should be familiarized with the response procedure. The last step of the response procedure is to alert the response team, so it is necessary for employees to also be familiar with the structure of the response team. No matter how much planning we do, we cannot ensure the success of the response unless we test the procedure under real circumstances. It is not advisable to wait until an attack occurs to try the response procedure. If is instead advisable to simulate an attack, and put the response to the test, this way the employees, as well as the response team, will have some experience when a real attack occurs.

Employees should be familiar with all the policies and procedures established by the company; otherwise, the consequences of an attack may expand beyond those caused by the attacker. For example, in Part 5.5 we talk about the possible necessity to monitor employees' activities, as well as email, and web access during an investigation. Suppose that a policy that allows the company to do it, and establishes that the employee has no reasonable expectation of privacy, is in place. If the employee manages to prove in court that he did not know about this policy, then the court may rule that the employee still had a "reasonable expectation of privacy." So it is vital not only to establish policies and procedures such as those mentioned in this thesis, but also to make sure that all employees know about them.

PART 8

POLICIES: SPEEDING UP THE INVESTIGATION

8.1 Introduction

Once an investigation is being launched, whether it is a preliminary investigation such as the case mentioned in Part 6.4, or a full scale investigation done by a computer forensics expert, the main goal is to conclude it as quickly as possible for the following reasons among others: 1) an investigation interferes with company activities, especially since the equipment in question cannot be used, to avoid destroying or compromising evidence, 2) the problem needs to be corrected as soon as possible to prevent another attack, and 3) as time advances the intruder(s) has a greater opportunity to cover his tracks by destroying or compromising the evidence.

8.2 Prohibit Encrypted Files

It is advisable for a company to prohibit the presence of encrypted files in its system unless authorized. Encryption is used when we do not want anyone to have access to the content of a file. Decrypting a file is a very difficult and time-consuming process, if there is no key. If the key is long enough, it may take months and even years to break it and reveal the content of the encrypted file. Discovering, and revealing the content of encrypted files is one of the first steps of a forensic investigation. During this process, we cannot count on the owner to give us the key to decrypt the file, so it is wise to avoid this situation. In addition to that, there are also legal implications when decrypting someone else's file. "Invasion of privacy," which we discussed in Part 5.2, is one of these issues.

8.3 Prohibit Scrubbing Tools or Shredding Software

Deleted files are also a source of evidence. The process of recovering deleted files is usually not that difficult or time consuming. However, it can be made very difficult and time consuming when certain type of software is used in the areas of memory containing the deleted files. This type of software includes scrubbing tools and shredding software, which are programs designed to destroy information. They wipe clean the targeted space by writing over clusters[8] several times. In some cases, even after the clusters are overwritten several times, the data or at least part of it can still be recovered; however, the time spent in data recovery increases greatly. So, in order to avoid unnecessary delays during the recovery of deleted files, it is advisable for the company to prohibit the use of this kind of software, unless authorized.

---

[8] Cluster: "A group of disk sectors (smallest unit that can be accessed on a disk.). The operating system assigns a unique number to each cluster and then keeps track of files according to which clusters they use [AOLC96]."

8.4 Data Indexes

The first step of an investigation is to check all the potential evidence collected. This is easier said than done, especially since extensive logging produces a great volume of data. The amount of time that takes to search through each entry in a log file is proportional to the amount of time necessary to go through all the books in a library one by one. But, when we are in a library, we use catalogs. We search by indexes and once we find the book we want, we lookup it up in the shelves. A similar concept can be used to cut down the time necessary to inspect all the potential evidence. The idea is to create an index of the data kept in the logs for investigative purposes only, and refer to the original information when necessary. One way to build this kind of index is by keeping summaries of the log data collected.

The Internet gets faster every day, and millions of packets flow across the network. As a result, the log files contain massive amounts of information about packet flow. In order to search through these logs faster during an investigation, we create summaries that include the most relevant information from each packet, and use these summaries as indexes. However, we face the problem, of knowing what to keep. If the information needed was not stored in the data summaries, then it is necessary to go through the complete log files to do the search, which defeats the purpose of having indexes at all. Examples of the type of information we may want to include in the summaries are the date, source/origin, destination, service port, and duration of TCP connection occurring on the network, the URL from every web request, the origin and destination of SMTP sessions, and the user-id from all Telnet, FTP, and rlogin sessions

[RANU97].

8.5 Information Fusion

Along with logging files, during an investigation it is necessary to look at the output of Intrusion Detection Systems (IDS). IDSs monitor the activities in an environment and then determine if the activities represent an attack, or are a legitimate use of the environment. An IDS produces indications and warnings (IW) values that must then be analyzed to determine if an attack has actually occurred. The output from the IDS is also a considerable amount of data, especially since, as we said in Part 5.5, we want to keep all of the output, whether an attack is detected or not by the system.

One way to speed up the inspection of this output is to use information fusion techniques to correlate all the values of warning and indication. This allows the investigation to go much faster, since the investigative team can look at the entire representation of the events that occurred, instead of bits and pieces from everywhere and trying to figure out how they tied together. Some of the techniques used to fuse information are single-object, and multiple-object information fusion. Single-object fusion, given IW values for a component from different intrusion detection techniques, determines a composite IW value for the component [YE98]. Multiple-Object Information fusion, given composite IW values from multiple components, determines a system-level IW value [YE98]. Either one of these techniques can be used depending on the kind of output from the IDS.

8.6 Anonymity

Anonymity is a great tool to protect privacy. It can help systems to resist traffic analysis, eavesdropping. It prevents an attacker from knowing who is communicating with whom. The attacker only knows that communication is taking place, but not the source or destination parties. However, for the purpose of an investigation, we need to know when, how, and who was in the system, which is very difficult to do if anonymity is allowed. So, on the one hand we know that anonymity is needed to protect the privacy of the company on the web, and on the other hand, we know that anonymity should be avoided within the company system if we want to be able to determine the identity of an attacker. We need to find a balance, and one possible solution is to use onion routers.

"The Onion Routing research project is building an Internet-based system that strongly resists traffic analysis, eavesdropping, and other attacks both by outsiders (e.g. Internet routers) and insiders (Onion Routers themselves). It prevents the transport medium from knowing who is communicating with whom -- the network knows only that communication is taking place [OR99]." The good thing about this system is that it can be installed at different points between the network and your system. It can be installed in each of the user machines, in the router, in the firewall or at some point in the Internet. For the type of anonymity we want to achieve, the firewall will be the best place to install the onion router, since it would allow you to see everything that goes on behind the firewall, but it will still provide anonymity for the company outside the firewall (Figure 7).

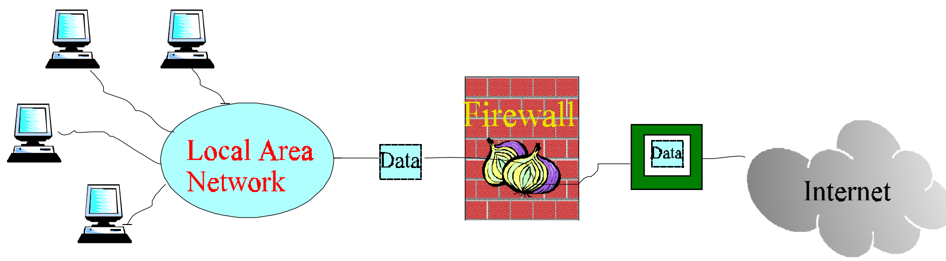

Figure 7: Onion Routing Example

8.7 Date, Time, User Stamps in Files

One last policy in this section is to keep time, date, and user stamps in all files, automatically. During an investigation, time, date, and suspect are three key elements. When an investigation is in progress, the investigator needs to know what date a file was created, or modified, or deleted, and who did it. This is a key point to be able to determine what happened exactly.

Part 9:

POLICIES: PROTECTING THE EVIDENCE

9.1 Introduction

Protecting the evidence is a key step in computer forensics. In order for this evidence to be useful, we must be able to prove its integrity. Part 7 ensured the training of personnel that may come into contact with the evidence if an attack occurs, but that is not enough. Up to this point we have considered the damage the data could suffer by being handled improperly, but we must also consider the damage the data might suffer while stored in the system. The damage, for example, could be caused by an attacker trying to destroy evidence of his crime, or an employee trying to erase incriminating data from the log files.

9.2 Password Protected

The evidence gathered should be protected at least with a password. However, password protection alone may not be enough to guarantee the security and integrity of the data. As we mentioned in Part 2, passwords can be broken using password cracker software, so they are not very reliable. It is preferable that we use encryption as described below.

9.3 Encryption

"Encryption is the most effective way to achieve data security. To read an encrypted file, you must have access to a secret key or password that enables you to decrypt it [AOLC96]." Potential evidence such as log files, IDS output, and the data indexes should be encrypted. In Part 8.2, we talk about prohibiting the presence of encrypted files in the system unless authorized. We established this policy because an encrypted file may contain evidence that we cannot access without a key, and if the file belongs to the attacker, he is not going to give us the rope to hang him. We can try to break the encryption, but this process could take years, so it is better not to have to worry about encrypted during a forensic investigation. This would be one case where having encrypted files will actually help the computer forensics expert because a key to decrypt the files would be available, and the process guarantees the security, and therefore, the integrity of the evidence.

9.4 User Authentication

No unauthorized access to the system should be allowed. Whenever a user tries to connect to the system, we have to make sure that it is a valid user. Allowing anyone to enter the system without questions could cause major disasters. That person may be trying to log in to erase all the files in the system, maybe to cover up something he did. In addition, user authentication also helps the last step of the forensic investigation, which is to identify the attacker(s). Knowing who is in the system at all times, makes this step much easier.

9.5 Access Control

Authorization is establishing permission to access a resource, such as a file. One way to establish authorization is through access control. Access control is a mechanism for limiting use of resources to authorized users [KAUF95]. This process establishes a relationship between users and files. We can establish the permissions on each file, specifying which users have access to the file, or we can establish the permissions on the users, specifying which files each user can access. This policy provides a start point for the investigation, since we know that if an attacker modified a file, it had to do it through one of the people that had permission to access that file. Instead of checking every employee to find a start point, we search the employees that have access to the particular file.

9.6 Integrity check

Just protecting the data is not enough. We might need to provide proof that the evidence has not been corrupted, so need to do periodic integrity check on the data collected. There are different techniques to do integrity checks, for example there is a technique that uses hash functions. A hash, also called a message digest, is a one-way function that always produces the same unique output for each unique input. This type of function is called one-way because it is very impractical, and sometimes impossible to compute an inverse for it. So, how do we use this to provide integrity check? First we feed the data collected to a hash function and we store the hash result along with the data. At the time the integrity check is needed, we feed the same data to the same hash function, and if the hash output is exactly the same as the one we had stored then the data has not been changed. However, if the output of the hash is not the same, then we know this data has been changed and it may no longer be useful.

CONCLUSION

As people get more and more comfortable with computers, and technology advances, society becomes more computer-dependent. In an era where everything from the stock market to air traffic control is managed through computers, security becomes a survival issue. In today's society, computer crime is a serious problem. Preventive measures are not enough anymore, we must find a way to catch and prosecute computer criminals, and computer and network forensics is the gateway to archive it.

However, we should not leave everything to computer forensics experts. If we are ever going to find a solution to the computer crime problem, it will be through a collaborative effort. Everyone from individual users, to company owners have to get involved. This thesis proposes policies to enhance the forensics of computer security by helping experts in the field do their job faster and more efficiently. Now, it is up to the companies, and users to adopt these policies according to their specific needs.

.

BIBLIOGRAPHY

ACF00 "About Computer Forensics," Technology Inc, May 7, 1999. http://www.computer-forensics.com/about/welcome.html

ANDE99 Anderson, Michael, "Computer Evidence Processing," Technologies, Inc. August 11, 1999. http://www.forensics-intl.com/art10.html

AOLC96 AOL COMPUTING'S WEBOPEDIA, AOL 1996. http://aol.pcwebopedia.com/

BURN98 Burns, Stuart, "Chasing shadows," *Accountancy*, vol.121, no.1254, p. 40.

MUFF95 Muffet, Alec, "WAN-hacking with AutoHack-auditing security behind the firewall," Proceedings of the Fifth USENIX UNIX Security Symposium, p. 223, 21-34, 1995.

CE90 "Computer Evidence," *The Computer Law, and Security Report* 1998-90.

CEP99 "Computer Evidence Processing," New Technologies Inc., April 2000. http://www.forensics-intl.com/art5.html

CEPS99 "Computer Evidence Processing Steps," Technology Inc, May 7, 1999. http://www.forensicS-intl.com/evidguid.html

CF98 "Computer Forensics," *SC Magazine*, October 1998. http://www.infosecnews.com

CIRG99 "Computer Incident Report Guidelines," Technology Inc, May 7, 1999. http://www.secure-data.com/guidelns.html

CIVI98 Civie, Victor and Richard Civie, R. "Future technologies from trends in computer forensic science," 1998 *IEEE Information Technology Conference, Information Environment for the Future*, 1998.

COLD98 Coldwell, Roger A., *Hacking into computer systems, anomie and computer education*, Computer Education, no.90, p. 20-3, 1998.

COLL90 Collier, Paula and Barry J. Spaul, *Information System Forensics,* Journal of Information Technology, 1990.

COLL92 Collier, P.A. and B.J. Spaul, *A forensic methodology for countering computer crime*, Artificial Intelligence Review, vol.6, no.2, p. 203- 15.

CUHD00 Computer User High Tech Dictionary, ComputerUser, 2000. http://www.currents.net/resources/dictionary/

DAY97 Day, S.P. and Ford, S. *Hard Evidence from Computers*, European Conference on Security and Detection - ECOS97 Incorporating the One Day Symposium on Technology Used for Combating Fraud.

DEPO97 DePompa, Barbara, "Firewalls Deter Outside Attacks," Internetweek, September 7, 1997. http://internetwk.com/supp/security0901-1.htm

DEVA99 Devar, Hervé, Marc Dacier, and Andreas Wespi, *Towards a taxonomy of intrusion-detection systems*, Computer Networks, vol.31, no.8, p. 805-22, 1999.

DOWN98 Downey, Jeff, "Can't say no," PC Magazine, April 21 1998, v17, n8, p. 203.

DWAN95 Dwan, Berni, "Why the Cuckoo's Egg might never have been written!," *Computer Fraud & Security Bulletin*, p. 15-18.

EARL91 Early, John, "Who's been hacking into your computer?," *ICL Today*, March 1991.

EF99 "Electronic Fingerprints," New Technologies Inc., April 2000. http://www.forensics-intl.com/art2.html

FAQN00 "FAQ: Network Intrusion Detection Systems," Version 0.8.3, March 21, 2000. http://www.robertgraham.com/pubs/network-intrusion-detection.html

FELT97 Felten, Edward W. Dirk Balfanz, Drew Dean and Dan S. Wallach, *Web spoofing: an Internet con game*, Software World, vol.28, no.2, p. 6-9, 1997.

FERB97 Ferbrache, David and Sturt Mort, *Malicious Software and Hacking*, Information Systems Security, vol.6, no.3, p. 35-54, 1997.

FLUS98 Flusche, Karl J., *Computer crime and analysis of computer evidence: it ain't just hackers and phreakers anymore!,* Information Systems Security, vol.7, no.1, p. 27-32, 1998.

FRAN99 Frank, Diane, "Bringing it all together," *FCWonSecurity*, March 29, 1999. http://208.201.97.5/ref/hottopics/security/together.html

FURN99 Furnell, S. M. and M.J.Warren, "Computer Hacking, and Cyber Terrorism: The real Threats in the New Millennium," *1999 Elsevier Science Ltd*, 1999.

GARF99 Garfinkel, Simson L., "The documents in the case," *Technology Review(Cambridge Mass)*, July-August 1999.

GREE99 Greenlaw, Raymond, and Ellen Hepp, *Fundamentals of the Internet and the World Wide Web,* MacGraw Hill, California, 1999.

HOSM98 Hosmer, Chet, "Time-Lining Computer Evidence," *1998 IEEE Information Technology Conference*, Information Environment for the Future, 1998.

JOHN98 Johnson, Neil F. and Sushil Jajodia, "Steganalizys: The Investigation of Hidden Information," *IEEE Information Technology Conference, Information Environment for the Future*, 1998.

KAUF95 Kaufman, Charlie, Radia Perlman, and Mike Speciner, *Network Security*, PTR Prentice Hall, New Jersey, 1995.

LETI00 "Law Enforcement and the Internet," New Technologies Inc., April 2000. http://www.forensics-intl.com/art4.html

MARK00 Markoff, John, "CYBERPUNKS," *Upfront*, March 27, 2000. http://www.nytimes.com/upfront/

MCCL99 McClure, Stuart, Joel Scambray and George Kurtz, *Hacking Exposed*, McGraw-Hill, California, 1999.

MSDS00 "M-SWEEP Data Scrubber," New Technologies Inc., April 2000.

http://www.forensics-intl.com/ms.html

OR99 "Onion Routing," Onion-Info, July 20, 1999. http://onion-router.nrl.navy.mil/

QUIT95 Quittner, Joshua, "Cracks in the net," *Time*, February 27 1995, v145, n8, p34.

RADC98 Radcliffe, Deborah, "Handling crime in the 21st century," CNN.com, December 15, 1998.
http://www.cnn.com/TECH/computing/9812/15/cybersleuth.idg/

RANU97 Ranum, Marcus J., *Network Forensics and Traffic Monitoring, Computer Security Journal,* Computer Security Journal, Volume XII, November 2, 1997.

ROBB98 Robbins, Judd, "The Devil's Advocate: How Computer Forensics can support both sides of computer litigation,"*AALL Spectrum*, March 1998.

ROSE97 Rosenoer, Jonathan, *CyberLaw,* Springer-Verlag, New York, 1997.

RUTE00 Reuters, "Hackers got Bill Gates' credit card info," *ZDNet*, March 27, 2000.
http://www.zdnet.com/zdnn/stories/news/0,4586,2473689,00.html

SCHA77 Schabeck, Timothy A, "Computer crime investigation: the forensic approach," *1977 Carnahan Conference on Crime Countermeasures*, p. vii+243, 133-9.

WILD96 Wilding, Edward, "An Introduction to Computer Forensics," Proceedings of COMPSEC International 1996, 13th World Conference on Computer Security Audit and Control, 1996.

WILS97 Wilsker, Ira, "Internet for law enforcement a modern phenomena and a phenomenal tool," *Proc. SPIE - Int. Soc. Opt. Eng. (USA)*, 1997.

WTDI99 "What to do if the unthinkable happens." Vogon International, 1999.
http://www.vogon-international.com/investigation_services-06.htm

YE98 Ye, Nong, Joseph Giordano, John Feldman, and Qiu Zhong, "Information Fusion Techniques for Network Intrusion Detection," *1998 IEEE Information Technology Conference*, Information Environment for the Future, *1998*.

BIOGRAPHICAL SKETCH

Ever since I was a little girl, I have been interested in science. At first I was inclined to make my career in the Mathematics, but this change the first time I sat in front of a computer in 1990. Ever since that moment I have dedicated my free time to improve my computer skills. First I learned to work MS-DOS, then I learned different programming languages such as DBase, Pascal and some C. During this time, I was still in my native country, Cuba, and the resources were very limited, but I still manage to create a few projects during the course of the following five years. During this time, I developed a software to help people improve their Spanish skills. The program was developed in Pascal, and it included a series of exercises to help improve a series of skills in the Spanish language. In 1995, I move to the United States where I continued my high school education in Miami Senior High. When I arrive, I was placed in an AP Computer Class, where I developed simple projects. During my senior year, I got the permission of the school, and the help of my teachers to enroll in the AP computer class again, but with the purpose to develop a project of my own. This time was a computer-assisted program to help students improve their English through the use of the computer. This program teaches various concepts, such as verbs, adjectives, nouns and customs in the United

States. I donated this project to the ESOL department at Miami Senior High. In 1996, I graduated from high school, and I enrolled in Florida State University as a computer science major.

Ones in college, the curriculum of classes necessary to obtain a Bachelor Degree, allowed me to extent my interest to a lot of different areas within the computer field. Through my curriculum, I have learned to program in new languages, such as C++, and Assembly; I have acquired skills in fields totally new to me such as Databases, Computer Architecture, and Digital design. I have broadened my knowledge of Operating Systems, and Programming Languages in general. In addition to does skill acquired through the required curriculum, through my choice of electives classes, I got the chance to learn about areas of particular interest to me, such as Software Engineering, CyberLaw, The Internet, Graphical Design, and Computer and Network Security. While studying these areas, I have acquired knowledge, and experience of a few more programming languages, such as Java, HTML, Javascript, and CGI. I have also acquired knowledge about the internet such as: website's design, protection, and functionality.

In September 1999, my senior year in college, I decided to do honor in my major, computer science. When I first started in this project, I did not know what area I wanted to work in, but I knew whatever the area it had to meet three conditions: 1) it had to be an area heavily related to the internet 2) it had to be an area in development now, and 3) it had to allow me to pull together as much of the knowledge I have in the different areas of computer science as possible. So, after a while of searching I meet Dr. Yasinsac, a new professor in the computer science department. Dr. Yasinsac was interested in doing research in the area of Computer and Network Security, and after talking to him, I decided that would be an interesting area to do my research in. After a few weeks of research, we decided that I should write my thesis in the area of Computer Forensics.

For more information, about my work please visit my web site at: http://www.cs.fsu.edu/~manzano/ or email me at: manzano@cs.fsu.edu