1. INTRODUCTION

It is apparent that information technology is the backbone of many organizations, small or big. Since they depend on information technology to drive their business forward, issues regarding network security have become a high priority. Companies use technologies such as e-mail, web services, databases, applications, etc. – accessed via a network - on a day to day basis to perform various organizational functions and duties.

The task of protecting a company's network falls on a single or group of persons called Network Security Administrators. They use a wide variety of tools and procedures to keep the company network secure. One major classification of tools used are Intrusion Detection Systems or IDS. IDSs are capable of monitoring network traffic and system usage for anomalies (activities that are not part of the norm).

The aim of our study is to perform a hierarchical task analysis of functions carried out by a typical Network Security Administrator in a small to medium sized company. Hierarchical task analysis is the detailed description of a job or function from top to bottom. By performing this study, we are able to design better user interfaces for applications used for intrusion detection. Current practices involve using a variety of text based tools and utilities, monitoring log files, etc., all of which may be time consuming, monotonous, and error prone. A better user interface for the IDS would allow the Administrators to be more efficient and productive in their job and reduce the probability of error.

2. INTRUDER DETECTION SYSTEMS

Intruder Detection is the process of detecting inappropriate or harmful activities in a network infrastructure. Intruders to a network can be external, or from within the organization. Recent studies conducted by the FBI and Computer Security Institute have found that more than 81% of intrusions have been committed from within the organization [1]. While organizations are more guarded against attacks from outside, they often neglect the danger within; damage done by inside intruders has the potential to be more severe. Therefore, while protection from outside intruders using securities like firewalls would be strong, inside network may be left vulnerable to attacks from an insider. The use of an Intrusion Detection System can help track down internal hackers, monitor them, and catch them in the act.

2.1 Types of Intruder Detection Systems

There are three major types of Intruder Detection Systems. They are:

- Host based Intrusion Detection System.
- Network based Intrusion Detection System.
- Hybrid based Intrusion Detection System.

2.1.1 Host based intruder detection systems

This IDS entails monitoring servers, application software, database servers, and so on. Monitoring is usually done by examining log files for unusual activities, and analyzing system processes, hard disk usage and critical system files such as password, network and server configuration files.

Host based IDS can be signature based or anomaly-based.

Signature based tools monitor well known signature and patterns of worms and scripts. *Signatures* are well known sequence of string or combinations of packet headers that match a known network attack. Such tools have to be kept periodically updated to new and evolving patterns.

Anomaly based ID systems, such as *tripwire*, use well known facts about the system such as timestamp, size of key system files, etc. This system will notice when an important file has changed and will alert the administrator if the change was unauthorized.

2.1.2 Network based Intrusion Detection systems

A network based Intruder Detection system monitors the network traffic for well known patterns generated by intruders. This is done by placing a network sensor to capture all traffic that flows through the network segment. A sensor can be a network device that does not alter network traffic, thus remaining invisible to the network, but at the same time reading all the traffic and analyzing packets for interesting patterns that match known signatures [2]. Network based IDS can also monitor ports for suspicious connections and watch for well known port attacks.

For example, a Unix command string such as *"rm -rf /"* can potentially delete a complete Unix file system or *echo "++" >$HOME /.rhosts* can allow any user from any host to log into the system without requiring a password. When a network based IDS senses such a sequence of string, it can take defensive action. For instance, the IDS can block out the intruder, or alert a security administrator. Some examples of network based Intrusion Detection systems are *AXENT (www.axent.com), ISS (www.iss.net) and CyberSpace*. A popular open source network based ID system is *snort (www.snort.org)*. *Snort* works both on Windows and Unix platforms. It comes with a set of rules that can trigger actions, and also allow customized rules.

2.1.3 Hybrid based Intrusion Detection Systems

Both Network and Host based IDS have their own unique advantages and disadvantages. Network based IDS are easier to deploy and are less expensive to purchase and maintain. However, their performance depends on known security exploits and signatures. If a new exploit is used that the IDS is unaware of, the system could easily fail to detect the attack. A host based IDS is only as good as the security administrator who maintains and monitors it. Becoming skilled at, maintaining and monitoring this software can be a daunting task. Therefore, the best approach is to use a combination of the best features of Network based and Host based IDS to improve resistance to attacks and to provide greater flexibility. This approach is commonly referred to as Hybrid IDS.

2.2 Types of network attacks

Some well known types of network attacks are listed below.

i. *Denial of Service or DOS attacks* attempts to deny an authorized user from using the system for productive use [3]. For example, sending a TCP SYNN packet which is constructed with the same source and destination IP addresses. This can potentially lock up the system and will have to be rebooted. Another common DOS attack, called the *Ping O' Death* attack, sends a ping request with an abnormally longer payload (over 64K bytes). This can cause a buffer overflow in older operating systems and can possibly lockup or reboot the machine [3].

ii. *IP address spoofing* is a method of constructing or spoofing IP packets so they appear to be from a trusted or known host. An intruder could gain access to internal servers by pretending to the server that the network traffic from the intruder is from a trusted system [10].

iii. *Sniffer attack* is carried out by using special applications that can read network traffic and extract interesting information such as passwords, credit card numbers or other sensitive data. Encrypting network data using digital signatures can prevent sniffer attack [10].

For an intrusion detection system to be effective, it should meet several basic characteristics. Some of the basic characteristics of a good Intrusion Detection System include the following.

- It must be able to protect itself from intrusion by monitoring itself. An intrusion detection system is of no use, if it can be compromised by an intruder.
- It should not make a big impact on the performance of the network or the hosts it is monitoring.
- It should be able to monitor the network for deviations from normal behavior or normal usage of a computing resource.
- It should be customizable for different types of organizations. For example, a university will need a different network security level than a bank. A bank requires a network policy that would make the network very secure to minimize threats to its application and minimize uptime. Universities will have a huge number of workstations in their network and students would require a more open network system, but at the same time protect important applications and databases.
- It should be rule based so an administrator can add or change rules which will allow enhancing the system to counter new security threats.
- If it's a network based IDS, it should be able to keep itself updated with the latest patterns of attack. This can be done by downloading and applying new rules automatically from trusted sources.

3. HIERARCHICAL TASK ANALYSIS

Hierarchical task analysis is the detailed description of a job or function from top to bottom. This is done by decomposing each step required to complete the job into greater detail. It shows the hierarchical relationships between the tasks and provides an in-depth understanding on how a job or function is carried out.

Our aim in this study is to perform a hierarchical task analysis on the use of various intrusion detection software used by a network security administrator. By studying the various tasks associated to the function of a network security administrator, we will be able to design an effective user interface for intrusion detection systems. A well designed user interface would allow administrators to meet their goals faster and more effectively.

The hierarchical task analysis were developed using the following steps:

- Observing the use of intrusion detection systems in a medium sized organization.
- Studying commonly used intrusion detection techniques in the industry.
- Interviewing network security specialists.

Seven major tasks were identified in securing the network in an organization. These tasks are explained in detail in their respective task descriptions. The seven major tasks are [4]:

1 Developing Security Policy
2 Selecting an Intrusion Detection System
3 Hardening and Secure Network and Servers
4 Preparing for Intrusion Detection
5 Detecting Intrusion
6 Responding to Intrusion
7 Improvement

3.1 Network Security Administrator

The *Network Security Administrator* is an individual or a group of people within an organization who are responsible for securing the network, servers, workstations, data and other IT related assets in the organization. An intrusion detection system can only be useful when proper precautions are taken to lock down the network, and by having network administrators who are vigilant, alert, and knowledgeable about the system and about users of the system. This can be compared to having a security system in your home but forgetting to lock the windows and doors at night and when away from home or having alarm signals that are not acted upon. It is also important for the organization to have a detailed security policy.

3.2 Functions of Network Security Administrator:

Some functions of a network security administrator or group of administrators are listed below.

- Develop, maintain and implement an IT security policy for the organization. Educate the management and stakeholders about the need to abide by the policies. Train, and get feedback from the users of the infrastructure.
- Implement and maintain a firewall that keeps unwanted traffic out of the network.
- Monitor the network and servers for unusual activity. This is done by auditing important logs in servers, workstations and routers. Special tools can be used to automate monitoring logs. For example, *"Logwatch"* is a popular log utility that produces customized reports of several important log files.
- Monitor critical system files. Tools like *"tripwire"* help system administrators keep check on the integrity of critical system files. Some examples of files that need to be monitored on a regular basis are password files, database and web server configuration files, private keys of servers, binary files of the operating system such as kernel etc. Intruders also try to replace critical binaries and install back door *Trojan horse*. A *Trojan horse* is a program that pretends to be a useful application, but is really is a destructive process that could allow an intruder

access to the system, or create harm to the system. An example could be a free anti virus software from an unknown web site that could actually be a program tainted with a virus that would cause damage to the system.

- Backups are an important part of a secure network. In the event that a server was compromised, the administrator should be able to restore the server from secure backups with minimal downtime, and loss of data. Tools like *tripwire* can take a complete snapshot of the system in a secure state. The administrators need to check daily backups, and make sure that the media is stored in a safe place.
- Network forensics will enable the administrator to find out exactly what damages are made to a system in the event of an intrusion. For example, consider that there are many servers running database, web and application servers. If an intruder broke into the one of the systems, the administrator should be able to quickly determine exactly which systems were compromised before more damage is done. In most cases, the whole network cannot be brought down to contain the situation. Forensics can help in isolating the intrusion and help contain such situations. This method will also halt an intruder before they do further damage.
- Secure servers and workstations by removing default operating system privileges, unused open ports, and so on. By limiting functionality to what is really needed by the user, the risk of compromise is reduced.