Intrusion Detection Systems

Gerardo Maya
April 16, 2003

Introduction

For the last 10 years, data networks have evolved greatly and radical as the technological developments have been, the challenges in the use of these networks have been even more dramatic. In the past, the use of data networking was confined to local area networks, primarily to share resources such as printers and fileservers. Large corporations might make use of wide-area networking to connect remote sites, but these connections were typically private networks. Even thought connections to public networks existed, they were typically used for the limited passing of information, such as email but without allowing access into the enterprise network.

This usage pattern began to change with the appearance of applications and ways of doing business that required opening the enterprise networks to "outside users", such as remote employees, customers, partners and suppliers. The World Wide Web (WWW) was not the first such technology but it has certainly had the most dramatic impact, in providing a powerful new communications tool for business-to-business and e-commerce applications, the web also created a strong incentive for organizations to provide access to "outsiders" and in turn, created a series of new technical and procedural challenges.

Information Security is one of the greatest challenges facing those who want to connect enterprise networks to public networks. The incremental use of data networks and its applications promote additional vulnerabilities, the same connection that provides legitimate access for employees, customers and partners, provides also an avenue for attacks into the enterprise network and the devices in it. The challenge facing enterprise managers today is to satisfy business needs by providing more access to more information, while maintaining the privacy and integrity of the enterprise network.

By the end of this document you will understand the way Intrusion Detection Systems (IDS) work, how they are composed and the advantages and disadvantages of the different types of IDS. Also, at the end of this document there will be a section that shows two analyzes on two Intrusion Detection Systems that display the result of this research paper.

Intrusion Concepts

Before analyzing an Intrusion Detection Systems it is necessary to understand, what an Intruder is. Webster defines intruder as an individual that "thrusts itself in without invitation, permission, or welcome", in other words an individual who gain access to devices and information that doesn't belong to him and to which he/she has no lawful rights. There are two types of potential intruders:

External Intruders: Are individuals that come from the "outside" of the network and penetrate the perimeter defense in an organization. Most people perceive the outside world as the greatest threat to their security. Almost everyone thinks of a "hacker" coming through the Internet as the main security problem.

Internal Intruders: These intruders are those individuals who have some type of rights to one or more network entities or devices in an organization and who exploit those rights to gain access beyond what they are authorized to do. Rarely, the "insider" intrusion is accidental. In fact, statistics from 1990`s that as high as 80% or intrusion are coming from the INSIDE of the enterprise! [5]

Intrusion Types

There are three primary ways that an intruder gets into a computer system or network:

- **Physical**: The intruder gets physical access (console access) to a critical computer on the network. This usually a matter of "physical" security because at times, all that the intruder has to do is to find a computer that is already logged on, or one that has the user name and password taped on the monitor!

- **System**: is a type of cracking where it is assumed that the intruder already has a low privilege user account, but uses known security holes in the server or network system to gain administrator privileges.

- **Remote**: The intruder penetrates a system remotely across a network the Internet is a good example. The intruder usually begins with no special privileges and attempts to gain privileges or do damage through exploiting security holes.

Experts define ANY intrusion as a threat, not to be ignored. The Intrusion has the potential of deliberate, unauthorized: Access to important enterprise information, manipulation of important enterprise of data and rendering of a system to an unreliable or unusable state.

The need for Intrusion Detection Systems (IDS)

A Firewall is the first line of defense for your network, but they should not be considered as "the only solution" by any means. No single product is. Firewalls provide a basic level of security when deployed on the network perimeters and throughout the infrastructure.

Almost all companies that are serious about their business have invested in and deployed firewall technology. The fact is that the firewalls are open to compromise and can be externally attacked or bypassed in a number of ways. For example an attacker can exploit firewall miss configurations, circumvent the firewall by dialing through the telephone line, launch Denial of Service (DoS) attacks on specific services, use Trojan horses and tunneling, and even launch buffer overflow attacks to gain root access on the firewalls. Because internal attacks account for over 70% of incident on a network [1], firewalls must be also deployed internally around critical networked assets to lower the risk associated with intrusion. Again these firewalls can be circumvented or exploited by internal attackers [16].

Firewalls can be considered as the gatekeepers of the network, but they are limited in the protection they deliver. Their biggest downfall is the fact that most firewalls do not inspect the content of the packets they pass. To inspect the contents of these packets, your company must add an intrusion detection layer to your security implementation. IDS systems help identify the attack at en early stage, providing organizations with faster incident analysis and more time to respond to the incident and deploy mechanism to prevent further occurrences.

How does Intrusion Detection Systems Works?

Intrusion Detection System (IDS) are complementary solution to firewall technology. An IDS that has sensors both inside and outside the firewall can help determine whether the firewall is configured and operating properly. An IDS also recognizes attacks against the network that firewalls are unable to see, because the attack does not pass through it.

Diego Zamboni [18] defines that every Intrusion detection systems is desirable to have the following characteristics:

- The IDS must run with minimal human supervision.
- The IDS must be fault tolerant
- The IDS must being able to recover from crashes
- Must be able to monitor itself, to detect if has been modified by an attacker
- Must give a minimal overhead to the system where it's running.
- Must be scalable to user changes, for example new applications installed, and also must be scalable to support a large number of entities (hosts).
- Must support user configuration without the need to restart the system

Regarding this characteristics IDS may fall into four main categories:

- Traditional Network Based IDS (NIDS)
- Traditional Host Based IDS (HIDS)
- Hybrid IDS
- Deception System

Traditional Network Based IDS (NIDS)

NIDS works by using network cards in promiscuous mode, looking at every packet that passes on the network, trying to find a pattern that represents an attack. A typical network IDS consists of one or more sensors and a console to aggregate and analyze data from the sensors. Sometimes the deployment is easier and more manageable than a Host based IDS

[1] CSI/FBI 2000

solution, but once installed, some Network IDS miss some attacks because they can not keep up with high volumes of network traffic and/or they generate an unmanageable number of alerts due to false positives, making a real attack difficult to identify [4].

False positives are alerts that are generated due to legitimate activity, when in fact there is none attack taking place. There is a big problem with the false positives because when a company is repeatedly hit with false positives, they begin to ignore their alerting system and the data that it collects, rendering the system potentially useless. False positives are a constant challenge for most organizations; the approximate number of real attacks is one or two per 1 million of events [19].

The investigations made by Martin Arvidson and Markus Carlbark [17] reveal that NIDS aren't very useful for small companies with limited number of critical host like web servers or DNS servers. At these sites, it's more beneficial to deploy patches as fast as they become available, but in larger companies with many critical servers, it's very expensive and time consuming to maintain patches an all the server, these sites can gain a lot of benefits from NIDS because the will detect attacker trying to exploit server that have not been patched.

Traditional Host based IDS (HID'S)

HIDS by the other hand watch for processes inside the host and monitors log files and data for suspicious activity. When a process not defined is running or when the log file registers some activity an Alert is generated. Some host based IDS operate independently. In other systems, each host based IDS may report to a master system that centralizes the evaluation and response mechanisms, helpful in large enterprise deployments. On the other hand distributed systems permits more scalability sharing information between systems and reporting on a central console. As with most host based solutions, platforms availability and coverage make this a difficult solution to manage and allows systems to be open to network attack due to the lack of packet inspections capabilities [4].

Hybrid IDS

They combine host based IDS with network IDS technologies. Hybrid IDS are system based and provide attack recognition on the network packets flowing to or from a single host. Hybrid systems brings advantage over the NIDS so do not inspect every packet goes by, avoiding degradation issues of traffic analysis. Hybrid IDS provide additional protection by monitoring the systems events, data, directory and registry for attack. Again, platform availability and deployment problems are an issue and they are traditionally system resource intensive, but are less susceptible to false positives than network based IDS.

Deception Systems

Also known as "honey pots", provide additional level of security into a network. Deception systems data is usually more valuable due to the reduction of both false positives and false negatives because this kind of systems don't have activity due they are installed in order to catch any user entering data to them. Deception systems are suspect by nature.

How are intrusions detected?

IDS use a number of different technologies to detect malicious activity. The three most widely distributed technologies are signature detection, behavior anomaly detection, and protocol anomaly detection.

Signature based Detection

Also known as Misuse Detection, Knowledge based or Burglar Alarm. In Signature Detection, it is assumed that a signature could represent all misuse activity. In theory, even a slight variation of the same attack could be matched to a signature, very much like virus detection. An administrator would maintain a set or library of Intrusion Detection (ID) signatures that would be updated periodically. Almost all ID tools today are knowledge based or misuse detection systems.

PROS: If the set of ID signatures is kept up-to-date, many or all known ID methods could trigger an alert and the potential for false alarms is minimized.

CONS: An unknown or new ID method could penetrate the network. An expert must write and distribute signatures in a timely manner. Signatures must not match non-intrusive activity, to cause false positives. Also, this type of system is usually tied closely with the environment or platform on which it runs. This type of system may require multiple signatures to protect against a single vulnerability to be exploited. Also, this technique doesn't apply well to the stream-based traffic like http because is limited to inspect a single packet of the traffic

Behavior Anomaly Detection

Also known as Anomaly detection based or Expert Systems. In anomaly detection, all activity in a system is first marked as anomalous. Then, over the time, patterns of use are recorded. These patterns are recognized as normal activity profile. Once that profile is established, what is outside the realm of the normal profile is considered anomalous. To be effective, the established normal profile would have to be almost continuously updated, a task that results in unacceptable system overhead. Very few tools today use this approach.

PROS: This type of system could, in theory, be self-administering and self-regulating, without the need to subscribe to an updating service. Also it should be able to detect the new unknown attack. Too, it can detect internal abuses of privileges.

CONS: Anomaly detection is currently unreliable it is very difficult to model what is normal, so there is a high false alarm rate. Attempts are being made to drive it with an AI (Artificial Intelligence) engine, however, the normal usage of most networked systems is too varied and changeable to allow this to work in practice. Also, it is possible that if intrusion is in progress while the normal profile is being built, the normal profile will contain permissible intrusive behavior.

Protocol Anomaly Detection

Protocol anomaly detection is performed at the application protocol layer. It focuses on the structure and content of the communications. Many attacks target protocols such as Telnet, HTTP, RPC, SMTP and Rlogin for example.

When protocol rules are modeled directly in the sensors, it is easy to identify traffic that violates the rules, such as unexpected data, extra characters, and invalid characters. That is exactly how some of these attacks can be identified. Protocol based IDS, for example can detect code red, because they model the HTTP protocol exactly as it is reflected in the RFC 2616 [8]. The Code Red Attack violates the HTTP protocol specification because it uses a GET request to post and execute malicious code on the victim server. The IDS recognizes this as a violation of the protocol and alerts the system administrator to the violation. While the same kind of attack is making its way past signature based systems, this attack is recognized by the IDS as a protocol violation and is reported to the system administrators, giving them hours, sometimes even days to respond to the new threat before a signature for the attack is developed and distributed.

PROS: This type of system can detect attacks without user intervention. If an attack is used an anomaly in protocol, this tool could detect the attack by itself, the false positives are almost null because it only shows packets doing anomalies on the protocols and also it represents less overhead because signatures to new attacks don't need to be developed.

CONS: This type of detection has been used for a long time as a pure concept but it hasn't grown as technology and hasn't evolved to take a significant market share. This is because there are to many problems regarding protocol anomaly detection technology [17]. Also, the attacks that accomplish the rules of protocols could be pass thru this kind of system.

Combining Host Based and Network Based IDS

Combining host based and network based IDS provide greater protection that using either strategy alone. A network based IDS can often detect suspicious activity as an attacker probes the network to discover its resources before mounting an attack against a network. This early warning allows a security administrator to deal with the attacks proactively. A host based IDS, on the other hand, is better designed to detect more operating system and application specific attacks, based on system information that is not available to the network based IDS. In addition, a host based IDS can detect attacks that originate from the local console and that are thus undetectable by network based IDS.
Host and network based intrusion detection can be described as a set of methods for discovering attack signatures in network traffic and system data.

Host Based IDS

Host based IDS fall into two basic categories: single system and manager/agent. A single system protects one computer, by detecting intrusion in the machine's audit logs. Agents can remotely install/upgrade new versions and attack signatures “rules”. This type of configuration allows security administrators to define or distribute rules from one central location. A manager/agent “Host based” IDS application involves placing agents on each critical servers throughout the enterprise. These agents are connected to managers, which in turn are connected to a central management console for report and alert proposes.

When suspicious activity is detected, the IDS software can notify a console, send an e-mail, beep a pager, disable a user account, terminate the intruder’s process, terminate the intruder’s session, shut the system down, or execute a command. Some host monitors can also track audit from others applications, like firewalls, web servers and routers.

A simple example of this type of IDS is a web server monitoring system. First, web server log files are analyzed to syntax of the logged entries. Then sets of suspicious-activity rules are defined (e.g. a web root directory is being removed from a web-server, configuration is being altered, new accounts are created on the server). The IDS is then implemented to watch for those rules in the server log file.

Deployment, however, is highly dependent upon the IDS tool’s ability to consider the following three critical factors: scalability, platform support and process overhead.

- Scalability. The actual deployment and management capabilities of the complete IDS system. For example, “How many agents can be connected to a single manager, and how many managers can report to a single console?”

- Platform Support. How well the tool works in a heterogeneous environment becomes a critical factor when considering true “enterprise-class” IDS tools. Specifically, “How many different platforms does the tool operate on, and is there’s a cross-over problem when reporting between platforms?”

- Process Overhead. The balance between the overheads required capturing process communications or auditing OS and applications activity logs and the ability to react to infractions. An effective host-based IDS tool must have some sort of functionality by which its reporting mechanism can be *modified* so as not negatively impact system use.

Host based IDS are designed to monitor a single host on which the IDS agent resides. This kind of IDS is able to watch data available from higher levels of protocol stack, which restricts its ability to monitor activity to those audit trail made by OS or applications. It also can detect the activities that occur locally on the monitored host (e.g. file permission modification and user account setup).

The benefits of host based IDS are that they are easy to configure for individual servers and applications. They provide tailored security since they can monitor specific OS or application events and can enforce corporate policies. Only host based IDS can detect an intrusion that occurs through the local console. And only a host based IDS can enforce a user-based reaction policy when an attack is detected e.g. disable the user account, terminate user process etc.

Considerations include the potential computational overhead on mission-critical servers whose security is being monitored (since the IDS agent resides on the same server). Another consideration is the complexity of deployment and administration, which varies by the number and types of different servers being protected. Most importantly, host-based IDS cannot address attacks that exploit protocol vulnerabilities. And, since they analyze data from the audit trails, reaction to an attempted intrusion may not be in real-time but a good set of detection rules implemented became the IDS in a good reaction tool.

Network based IDS

The evolution of large networks requires monitoring data at all levels of communication. The main limitation of host-based IDS is that the access to audit trail is available only at the OS level or at the application level. This limitation led to the development of Network based IDS: Network based IDS forms its attack detection upon a comparison of parameters of the user's session and the user's commands to a rule-base of techniques used by attackers to penetrate a system. These techniques, called Attack Signatures, are what Network based IDS look for in the user's behavior. An attack signature can be any pattern or a sequence of patterns, which constitutes a known security violation. These patterns are then monitored on the network data. Since this model searches for pattern known to cause security problems, it is called a Misuse detection model. When such an attack signature is detected on any of the current user sessions, several actions can be taken to stop or trace the attacker – as well as record the event and notify an administrator.

The way that a Network based IDS works is by sitting in the middle of a fixed communication path between client and server has access to data at all layers of communication. Therefore this type of IDS can do extensive analysis for attack detection. Since the IDS is running on a system other than the server being monitored. The advantages of network-based IDS are: no performance overhead on servers of the application sessions (completely transparent and unobtrusive), and real-time detection and immediate response.

Network based IDS can be further divide in two sub-categories, one that utilizes a built-in attack signature database, called Static Signature IDS, the second which relies on signature information to be dynamically loaded into the ID, called Dynamic Signature IDS [5]. In the network based IDS model, each attack signature is processed using a set of functions, which essentially represent a program, to detect that specific signature. The actual Static Signature or Built-in Signature database engine is a collection of such programs (processing functions) one for each attack signature, which is built into the system. Drawbacks of traditional Static Signature IDS include:

- New attack signatures cannot be deployed in real-time because the IDS have no processing function for them – updates require the IDS system to be taken off-line.
- A new attack signature process has to be built and deployed in order to add new attack signatures – this negates timely deployment of new, vendor, supplied signatures.

- Vendor provided attack signatures don't address customer-specific applications.
- Performance is impacted due the sequential execution of each processing function of the attack signatures.
- Overhead increases as the number of built-in signatures increase.

To overcome the limitations of extensibility and performance of the traditional Static Signature IDS, a new technique called Stateful Dynamic Signature Inspection now for Symantec. Utilizing a knowledge base derived from Internet security and firewall implementations, developers designed virtual processing machine that allows attack signatures to be executed as a set of instructions. In this design each attack signature is a set of instructions, which the SDSI virtual processor executes using a cache entry describing the current user session state and the current packed received from the network. Each network server being monitored will have a small set of associated attack signatures based on only the operating system and application that pertain to that server.

Stateful refers to a virtual processor utilizing optimized register cache entries to store application state information for the current application session (builds application and command context about the monitored network sessions). This allows for efficient analysis and the recording of complex events. Dynamic refers to the real-time use of pre-defined attack signatures and the ability to readily deploy new signatures without taking the system off-line. Signature inspection refers to how the virtual processor compares cached session data to cached attack signatures, which are defined as a set of instructions (not unlike SQL constructs). This enables the vendor and the end user to define new attack signatures with minimal effort to address even the most sophisticated security violations. SDSI thus offers the latest technological advancements for intrusion detection systems. The advantages of Dynamic Signature IDS (Such as employed by Symantec NetProwler) include:

- Utilizes optimized attack signature customization for server security based on OS and Apps supported
- High performance hashed state-cache lookup
- Registered cache to store current info and instruction results
- Almost any type of attack signature can be configured using extensive instructions, which do not require programming
- New attack signatures can be added in real-time
- Customized attack signatures can detect attacks on customer-specific resources
- Vendor can release more timely updates

Network IDS is typically implemented on a critical segment-by-segment basis. Recommended implementation include:

a) Behind the firewall – all traffic from the internal LAN to the Internet and all traffic incoming to LAN, can be monitored
b) In a Services Network – which is more common with E-commerce protection of web applications; c) in front of key application servers – to efficiently defend against common and company – specific violations

c) In a server farm - to protect clusters of servers with mission-critical, transaction sensitive data on the same sub-network (common for financial institutes and ISP managed server farms);

d) Within remote work groups to protect sensitive information. Network Intrusion Detection system can only see traffic based on the segment on which they are installed and can only protect hosts that have static IP addresses. As long as the network IDS is placed on critical segments, it will be able to defend the most critical systems and applications.

IDS Performance

Network IDS typically placed in host NIC in promiscuous mode and caches network traffic for analysis because IDS does not use store / forward technology they does not impact network or application performance. Network IDS can only see traffic based on the segment on which they are installed. As long as the network IDS is placed on a critical segments, it will be able to defend the most critical systems and applications.

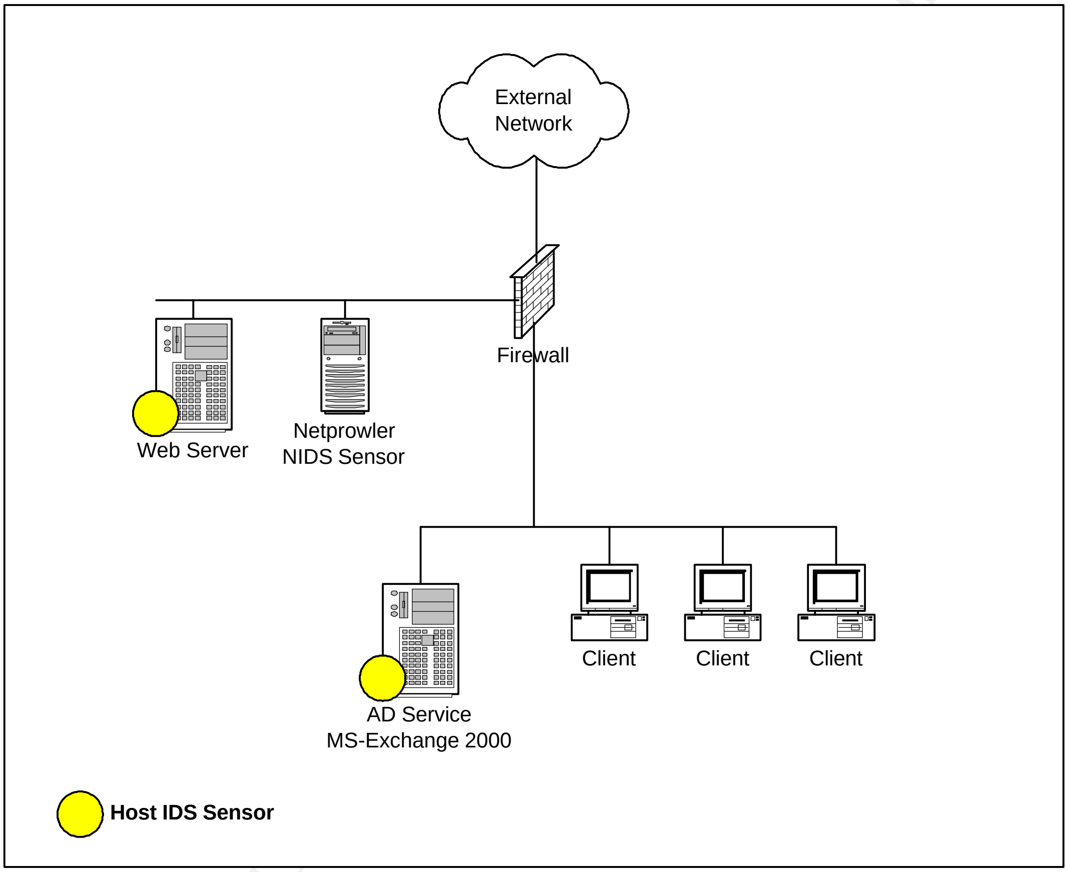
Network intrusion detection performance varies due to the speed of the network, the amount of traffic, the number of nodes being protected, the number of attack signatures employed and the power of the platform on which the IDS resides. Generally speaking, ID system may be overtaxed on busy networks. However multiple ID systems can be placed on a given segment to sub-divide host protection, therefore increasing performance and overall protection. High-speed networks, such as ATMs, which use packet fragmentation to improve bandwidth use, do pose problems in terms of performance and response. Most importantly, switched networks do not provide the static IP range needed for network ID systems. One method is to implement network ID systems on those segments with static ranges. One could also mirror switched ports through a switched management port. Another method is to bind the spanned ports into one to be monitored.

Host based ID systems complement network ID systems by supplementing protection often negated by switched environments. Host based IDS performance varies by the number of standard attack definitions and company-specific policies being monitored, as well as the number of processes being transacted on the given host. In addition to the speed of the host and its components, another factor is the architecture for host management.

Symantec Intrusion Detection Systems

There are several Intrusion Detection Systems over the Internet like snort, BlackIce and more but for a practical analysis I choose the Symantec products because the brand recognition and the good market position.

I setup a lab test environment to analyze the functionality and detection rates of the Symantec Intrusion Detection Systems solutions. The lab was constructed with the following configuration:



- A Windows 2000 server running IIS 5.5 fully patched configured as a Web Server
- A Windows 2000 server and Exchange 2000 supporting all the Active Directory (AD) accounts and the e-mail accounts.
- Windows 2000 professional workstations, working as a client.

In my lab, I configured a Services Network where my Web server resides; the server is protected using a HIDS in order to detect any bad action against it. I add a NIDS sensor to detect some attacks into the Web Server segment network. I used a HIDS into the Active Directory server for protect it.

Symantec Intruder Alert

I decided to install a Host based IDS in all my critical servers, after some analysis I found that the servers that need an HIDS were the AD server and the web server. I installed Intruder Alert into both systems, the process was quite simple and I could notice that the systems have had protection out of the box. Because of the architecture of Intruder Alert I installed it in a three-layer configuration:

- A manager who is the responsible of installs the policies to the agents and received all the alerts from them and sends the information to the console.
- An agent who is the piece of software that applies the policies given by the manager and generates the alerts when an event occurs.
- The console that is the piece of software where the policies are configured and the alerts are show.

I installed the manager into the AD server because or security reasons, the agents were installed into the both servers, I configured my firewall to pass all the information between the agent into a Network services and the manager into the internal network (I used the 5051 port).

Finally I installed the console into my workstation, I used the console to create some policies that allow me monitoring suspicious activity like creation and modification of accounts and the modification or elimination of the critical files into the web server.

I noticed that IDS work as a complement to my firewall, Intruder Alert enabled the development of precautionary security policies that prevent expert hackers or authorized users with malicious intent from misusing my systems, applications, and data. I set Intruder alert to provide complete control over my systems with policy-based management that help me to determine which systems and activities to monitor and what actions to take, as well I have real-time intrusion detection reports. I could use the administrative wizards to perform many routine tasks like silent installation and remote tune-up.

I configured Intruder Alert to monitoring the Web Server all the access to the systems as well as the files for the web page, when some file of the web page was modified or deleted the Intruder Alert sent me an alert and took an action. In this case the action was to restore the web page from a secure location.

After my work I conclude that the installation of Intruder Alert does not create overhead on either of the servers, it works fine with all the applications without a notable degradation of the system so I can say that Intruder Alert was not intrusive.

The capacity of detection was probed by doing dome actions like created and delete accounts, these action was notified to me in real time, access to critical files into the system were alerted to me also, I could check the bad logins into the servers too. I defined some policies for detect changes into the registry and it works fine too. Into the Web server the test was modifying a file, in this case Intruder Alert noticed the change and alerted to me at the same time Intruder Alert recovers the web page from the CDROM where I had a backup of it.

After the evaluation I considered that the best features of Intruder Alert are:

- Support for Most popular Operating Systems and application. Because of its broads platform support, Intruder Alert provides good coverage for enterprise networks. The Intruder Alert can be installed to protect systems running Windows NT, Windows 2000, UNIX (AIX, Digital/UNIX, HP-UX, IRIX, RedHat Linux, Solaris or Novell Netware. Intruder Alert knowledge of these OS allows detecting intrusion attempts such as failed logins, account changes and so on.
- Distributed components. The Intruder Alert architecture consist of three software components:
    - The agent a background process that monitors system events, looks for signs of intrusion attempts, takes action, and reports the events to the Manager.
    - The Manager, which serves as the communication link between the other components, as well as a repository for security events data, policies and configurations.
    - The Administrator, a graphical user interface that serves as Intruder Alert's administrative console for configuration and management of agents, policies, and users.
    - The Event Viewer, a graphical user interface that allows the administrator to view security event data and reports.

- Intruder Alert logical System Grouping permits make efficient administration; Intruder Alert organizes agents into domains. These are collections of Agents that are grouped together based on their operating system, location, workgroup, access restrictions or any other criteria that you designate.
- Role-based Security administration within Intruder Alert, you have the ability to set up administrative accounts with different levels of privilege. By adjusting the levels of privilege, you can define different security roles. This feature allows delegating security administration tasks, while still controlling your level of risk.

Symantec NetProwler

Once that I probed the features of a Host Intrusion Detection System the idea was probe the Network IDS. The main idea was to identify some attacks patterns getting into the Network services. I choose the Network Services to increase the security in that network segment.

I installed Netprowler into a NT workstation computer it was connected directly to the hub into the Network Services to analyze all the packets into the network services and matching them to the signatures database attack, when a packet matches Netprowler must notified and discard the packet. The attacks are categorized into a low, medium and high priority.

I change the category to some attacks in order to create alerts in the easy way. I change the configuration for the ping sweep attack to recognize five ping packets to the web server to recognize the attack immediately and alert it. With the installation and configuration of Netprowler I could validate that it instantly identifies, logs and terminates unauthorized use, misuse and abuse of computer systems by internal saboteurs and external hackers.

The types of alerts that Netprowler generates are SMTP messages, SNMP traps and pager alerts. The alert are configured by priority that's why I changed previously the priority to some alerts for probe it. I configured the SMTP alerts.

I applied some attacks in order to probe the Netprowler detection, the attacks was sent by another Vulnerability assessment tool called Symantec Netrecon. All the attacks were detected and only on these that were the high priority was set Netprolwer sent me an email. Doing a normal ping I could create an alert, it demonstrated too that modification and creation of custom signatures is quite easy.

The detection capacity was good, because I could detect all the attacks that I sent. These attacks were terminated by closing the session and dropped the packages. The creation of new rules was easy and the configuration too.

Other features that I could recognize are:

- Detection of hundreds of common OS and application attacks in real-time
- Network Profiling for "out-of-the-box" installation and automatic configuration
- Comprehensive attack signature customization wizard to protect company-specific applications
- On-the-fly loading of updates and new attack signatures while keeping defenses on-line and current
- Integration with Symantec Intruder Alert for enterprise monitoring of network and host security events.
- Employs SDSI Technology for efficient performance

NetProwler as Intruder Alert uses the 3 layer technologies, it permits that a manager can receive information for few NetProwler sensors. Again the console permits consolidate all the information collected by the sensor or agents and produce a great number of reports and queries.

After my review to this product I consider that the most important features about this products are:

- The distribute architecture, the use of the agent (sensor), Manager and Console components. This architecture provides centralized management, better network coverage, and enterprise scalability.

- Netprowler uses a process called profiler, which scans the network for live systems and adds them to the lists of systems to be monitored. This ability to auto discover and auto configure systems ensures better coverage, and thus better monitoring.

- Automatic application of attack signatures. As the profiler discovers new systems, NetProwler automatically assigns each system a set of attack signatures, those signatures permits to Netprowler recognize when an attack is taking place. By automatically applying attack signatures, new systems are protected by NetProwler as soon as they are discovered.

- The agent status monitoring that keeps track of the status of all agents. Alerts are sent when an agent starts or stops, ensuring that a segment will not go unprotected because an Agent went down and was not noticed.

- The Stateful Dynamic Signature Inspection allows to NetProwler to detect and prevent sophisticated attacks that are performed using a series of packets, each on which may seem harmless when taken out of the context.

Conclusion

As conclusion Misuse recognition of attack is the technique used by almost all the IDS nowadays, this preference over Anomaly is based on the easiest implementation and for the minimal false positive alarms. In the near future the tendency should be use both of the model for a better detection using the best of each model.

Based on the result of this paper a good intrusion detection system IDS should address the following issues:

1. It must run continually without human supervision. The process not be intrusive with the application and communications. It would be reliable enough that it may run in the background of the system being observed.
2. It must be fault tolerant in that it can survive a system crash and will not need its knowledge base rebuilt at restart.
3. It must resist subversion- that is; it can monitor itself against Intrusion.

4. It must not put a burden on the system being monitored. There should be little or no noticeable effect when running the IDS.
5. It must be able to observe deviations from what would be considered normal behavior.
6. It must have great scalability to bring to the enterprise the confidence to growth and still protected.

Finally a good implementation of “defense-in-depth” combines both host and network based agents to protect both protocol and OS and application attacks.

Works Cited

[1] Merriam-Webster Online
URL: http://www.m-w.com/netdict.htm
[2] Mark Crosbie, Katherine Prize. COAST Project “Intrusion Detection Pages”.
URL: http://www.cerias.purdue.edu/coast/intrusion-detection/welcome.html
[3] Robert Graham March 2000
URL: http://www.robertgraham.com/pubs/network-intrusion-detection.html
[4] David “Del” Elson. “Intrusion Detection, Theory and Practice”. March 27, 2000
URL: http://online.securityfocus.com/infocus/1203
[5] Paul Inella, Oba McMillan “An introduction to Intrusion Detection systems” Dec 6, 2002
URL: http://www.securityfocus.com/infocus/1520
[6] From Computer Security Journal vol. XIV, #1
“CSI Roundtable: Experts discuss present and future intrusion detection systems”
URL: http://www.gocsi.com/roundtable.htm
[7] Intruder Alert Fact Sheet
Symantec Corporation
URL :http://enterprisesecurity.symantec.com/content/promotions.cfm?PDFID=25&PID=11392085&EID=0
[8] Hyper Text Transfer Protocol /1.1
http://www.w3.org/Protocols/rfc2616/rfc2616.html
[9] NetProwler Fact Sheet
Symantec Corporation
http://enterprisesecurity.symantec.com/content/promotions.cfm?PDFID=29&PID=11392085&EID=0
[10] Training “Intrusion Detection Beginners”,
Symantec Corporation Apr 2000
[11] Training “Intrusion Detection Intermediate”,
Symantec Corporation Apr 2000
[12] NetProwler 3.5 User’s Guide,
Symantec Corporation
[13] Intruder Alert 3.6 User’s Guide,
Symantec Corporation
[14] Symantec Enterprise Firewall Administrator Student guide
Symantec Corporation
[15] The Big Picture
SANS Institute
[16] Managing Intrusion Detection Systems in Large Organizations, Part One
URL: http://www.securityfocus.com/infocus/1564
[17] Intrusion Detection Systems –Technologies, Weaknesses and Trends, Martin Arvidson
Markus Carlbark Stockholm 2003
[18] Doing intrusion detection using embedded sensors Diego Zamboni, October 18, 2001
[19] IDWG – IDMEF specifications, http://www.silicondefense.com/idwg/, January 2003