

Network-Based Intrusion Detection Systems in the Small/Midsize Business

Daniel Owen

spam@danielowen.com

<http://www.danielowen.com>

Abstract

This paper reviews the current state of Intrusion Detection Systems (IDS) with a particular emphasis on Network-Based Intrusion Detection systems (NIDS). Many of the topics covered will be applicable for any size business, but issues specific to the Small/Medium Business (SMB) sector are emphasized. The paper covers what an IDS is followed by implementation issues that should be considered when considering an IDS solution.

Introduction

Over the last several years the arms race between network managers and computer criminals has escalated. Through this escalation we have seen improvements in both the tools used by legitimate computer professionals and computer criminals. The first major improvement in network security was the firewall. This is still the standby of many networking professionals, but in today's hostile environment it is not a sufficient sole line of defense for any but the most unimportant networks. The Network-Based Intrusion Detection System (NIDS) has stepped into this void to help determine what attacks are not being stopped by the perimeter firewall thus allowing network and security professional to take appropriate counter measures.

Intrusion Detection Systems (IDS) are an outgrowth of auditing. Many systems are capable of creating audit logs. The difficulty is correlating these distributed logs into real-time or near real-time alerts. IDSs that monitor a single system using logs or add-on software are still popular and are more precisely referred to as Host-Based Intrusion Detection Systems (HIDS.) (Bace, 2000)

A NIDS is a system that listens to all the traffic on a given network segment. In doing so it is able to see both legitimate and attack traffic as it passes. Legitimate traffic is ignored while traffic that appears to be an attack will cause an alert. The nature of the alert will vary between attacks and between different NIDS softwares. (Crothers, 2003) I will look further into the details of how this is implemented and the challenges involved throughout this paper.

Modern HIDS typically add software loaded on the protected system that monitors for attacks. Tripwire is a classic example of HIDS software. HIDS come in

several forms but they all have disadvantages that make them less desirable than a NIDS solution for overall network security. HIDS software must be installed on every system that will be monitored. Since each individual machine must be monitored the overhead for a central system that correlates data can be substantial with even a relatively small number of monitored systems. (Bace, 2000) The final objection that I have to HIDS when looking at it as an either or solution in comparison to NIDS is that some systems will inevitably not be protected. There can be numerous reasons for this. A common reason for non-monitored systems is that the system is considered non-critical which still does not mean it can't be a starting point for further attacks against other more important systems. Many systems cannot run HIDS because they do not have the available spare resources available to run a HIDS. For these and other reasons I will concentrate on NIDS solutions in this paper. I do encourage the reader to investigate HIDS for critical systems as a further line of defense after installing a NIDS. I also highly recommend installing a HIDS on firewalls and NIDS even if no other systems are protected with a HIDS.

This paper is primarily aimed at the Small/Medium Business (SMB). SMBs are typically defined as being smaller than four hundred employees but larger than a Small Office Home Office (SOHO) implementation. Anything under about 50 employees is typically defines as a very small business. Much of what I will discuss will be relevant for any size business. To some extent, certain businesses due to their risk factors need to put themselves in the place of a larger business. For example a SOHO that has substantial intellectual property may need to spend more money on security than the office down the road that employs many more employees. Both of these companies may have found the

correct level of security for their risk factor. These are issues that can only be determined by the people inside the company who know their risk factors and what needs to be protected.

It should be mentioned that money is always an issue when it comes to security. It is often difficult for people outside the computer industry and sometimes for people inside the industry to understand the importance of security when they have not experienced a major security breach in the past. This type of thinking leads to complacency that can be dangerous. We must all remember that everyone who has a computer attached to a network has something worth stealing even if it is only CPU cycles and network bandwidth.

What does an IDS do?

Let's look first at how an IDS works. An IDS listens to all the traffic that crosses a network segment. So the IDS is first and foremost a network sniffer or if you prefer a network protocol analyzer. The IDS then looks at the traffic content to determine whether it is normal traffic or a possible attack. If the traffic is suspicious, the IDS alerts a human who in turn reviews the alert and decides what if any actions should be taken to eliminate the danger posed by the traffic that caused the alert. (Crothers, 2003)

IDS and Firewalls

This behavior differs from a firewall in that a firewall looks only at ports or services to decide whether to allow traffic to pass. As such anything traveling on an allowed port will be allowed to enter the protected network. If the traffic is for example a buffer overflow aimed at gaining root access on a vulnerable web server the owner of that

server may never know that they have been compromised until it is too late. This is where an IDS comes into play. An IDS can be looked at as keeping the firewall honest. It does not take the place of the firewall. For that matter an IDS does not block traffic it only alerts a human when potentially dangerous traffic is seen. Put another way an IDS is a layer of defense in depth. The firewall blocks traffic destined for ports that do not have legitimate public services on them, and the IDS alerts when something potentially dangerous is seen using one of the ports that is allowed through the firewall. An IDS can also be configured to alert when malicious traffic is seen within the LAN. As an example, a network aware virus will often try to attack other computers on the local LAN as well as on the Internet. An IDS should trigger on this behavior while the firewall will simply block or allow traffic depending on the port the traffic is traversing.

IDS technology is not a replacement for a firewall. Firewalls are designed to block very broad ranges of traffic. For example a firewall in front of a web server might block all traffic except HTTP, HTTPS, FTP and SSH. This means that even though the server may be vulnerable to a buffer overflow in a DNS service on the server it is protected from anyone outside the firewall exploiting that vulnerability because the attacker cannot get to the vulnerable service. It must be mentioned that the majority of attacks come from within the local network, so a NIDS watching network traffic behind the border firewall is indispensable since it will detect against insider attacks aimed at seemingly protected systems. (Graham, 2002) On the other hand an IDS system allows all traffic through and only alerts when known dangerous traffic is seen.

There is a variation of IDS typically referred to as either Active Response IDS or an Intrusion Prevention System (IPS). These systems use IDS detection techniques to

block traffic that appears to be dangerous. I will discuss their advantages and disadvantages later in the paper, but for now that is sufficient introduction to allow me to contrast IPSs and firewalls. As stated previously firewalls block very broadly. IPSs only block what they see as being dangerous. The obvious disadvantage here is that in the case of the web server that has a vulnerable DSN service running if the IPS is not aware of the vulnerability the service will be available for exploit. On the other hand an IPS will block attacks against aimed at the HTTP service. The firewall will allow HTTP attacks through because they are part of the allowed traffic. Once again, an IPS is another layer in a layered security plan not a replacement for the firewall.

The SMB and IDS

The SMB has some specific challenges to consider when looking at implementing an IDS. First many of the commercially available IDS systems are aimed at large enterprises meaning that the price of these commercial packages can be prohibitive. That is not to say that there are not good commercial packages available in a price range available to the SMB, but unlike many areas of security such as firewalls where demand has created many less expensive products and encouraged old enterprise only companies to offer less expensive solutions many IDS vendors only have very large expensive solutions. For many companies, both small and large, Open Source Software (OSS) is a viable solution.

The second and possibly bigger issue for the SMB is manpower. IDS implementations have traditionally been fairly complex. The person looking at alerts needs to have a fairly complete understanding of the operation of a network at the packet level. This understanding will allow the IDS manager to review packet dumps to

determine whether an alert is legitimate or not. Secondly the IDS will take up time for review of alerts. This is especially true during the period of time right after implementation until the IDS has been fully trained. (Crothers, 2003)

Due to the fact that most SMBs do not have 24/7 network operations employees there will be times when the IDS will not be actively monitored. This can lead to long windows of opportunity for attackers. If this is not an acceptable risk the only options are outsourcing of monitoring or using an Active Response IDS that responds on behalf of the human that is not onsite. Active response brings a new set of issue that I will discuss later in this paper.

Choosing a NIDS

There are a number of factors that must be considered when looking at a NIDS solution. In this section I will outline some of the most important decision factor as well as provoking you to think about issues that will be relevant in your own environment.

One of the largest differentiators among IDS systems is whether they use a signature based detection system or an anomaly based detection system. Most people that deal with security understand how virus scanners work so I will use virus scanners as a metaphor. Modern virus scanners have two levels or security. First they have a signature based scanning method and secondly they have a heuristic scanner. These approaches are approximately analogies to signature based and anomaly detection in IDS systems.

A signature-based system has the advantage of being very tightly tuned to a specific vulnerability meaning it should have less false positives. An anomaly detection system will look for things outside of the norms. This can obviously create somewhat higher false positive rates due to the fluid nature of network traffic. It should be

mentioned that in signature based systems most signature writers will make the signature fit the vulnerability as loosely as they think is possible without creating excessive false positives. This allows the signature to still match similar but non-identical attacks. This loose signature writing can also lead to false positives if legitimate traffic unexpectedly matches the attack signature. (Crothers, 2003) Some false positives may not be avoidable in cases where attacks look very similar to normal traffic. In these cases the IDS manager must decide whether the attack's possible consequences outweigh the work involved in evaluating false positives manually.

Another issue that must be considered is whether to buy an all in one hardware black box or to build a server and then load either commercial or OSS on top of that. As I alluded to earlier, cost may play a larger role in this decision than it does in other security decisions where affordable appliances are available to the SMB market. Having said that, if the budget for an all in one solution exists this type of solutions does have advantages. First and foremost black box hardware solutions are typically easier to manage than solutions that rely on the purchaser building their own hardware and then loading software. This can turn into a long run cost savings when looking at total cost of ownership. In some cases an all in one solution can even be plug and play for initial setup. On the other hand some software, especially in the OSS world, pays for the extra work in initial setup time with exceptional flexibility. It must be emphasized that most of the time involved in using an IDS is the day to day management required after the system is implemented. Due to this truth it is important that the person managing the IDS be comfortable with the user interface regardless of the underlying IDS architecture.

When looking at IDS software there are three basic groups to look at. There are commercial solutions, a number of open source solutions and in between you have open source solutions that have commercial versions available. First look at the budget available. OSS may be the only choice available for a company trying to implement a NIDS on a minimal budget. If the budget is available to purchase a commercial product that still does not mean that a commercial product will be the best option. In the realm of commercial software a number of products actually use signature files borrowed from the open source Snort project. OSS can often go toe to toe with commercial software in features and reliability. On the other hand, many commercial products have a tendency to have a more polished feel in the user interface. That's not to say that there are not bad commercial product interfaces or that there are not very good interfaces for open source software, but as a generalization commercial software generally has an easier to manage user interface.

When looking at an IDS performance is a major consideration. Purchasing a machine that cannot keep up with the traffic on your network can lead to lost packets. If the device is passively sniffing traffic this can lead to missed attacks. As an example if a network is sustaining a continuous 20-megabit of traffic and the IDS can only handle 18 megabit of continuous traffic there is a 10 percent chance of missing any attack. If the IDS is located inline, as is typically the case for active response IDS implementations, an underpowered IDS can become a major bottleneck.

Vendors treat inline IDSs differently when it comes dropping packets. Some vendors will configure their devices to simply pass anything that the IDS cannot review. If you are considering an inline IDS you should ask your vendor whether the device will fail open

(allow traffic to pass) or fail closed in the event that too much traffic is entering the IDS. Failing closed is the more secure option but can lead to the IDS system causing a denial of service if it is overrun with traffic either during a normal spike in traffic or as part of a denial of service (DOS) attack. Similarly, if you are considering an inline IDS it is important to find out what will happen if the IDS fails due to a hardware or a software problem. Once again some products fail open while others fail closed. Most software loaded on top of a customer purchased server will not have an option for failing open since the hardware and software must both allow this behavior.

As with any new system that is added to a network, support must be considered. Support varies from Internet message boards for some solutions to 24/7 toll free support for many commercial systems. There are a number of companies that offer support contracts for open source security products. So this may be an option to consider in looking at OSS solutions. It should also be stated that support that comes with many commercial packages can be absolutely horrible so an investigation must be undertaken before making a decision based on support level. Support is not a one-size fits all issue. If a company feels that in the event of a failure you can afford to remove the IDS for a week while you are correcting the problems that have appeared then support is much less important than if there is a need for 24/7 uptime with 30 minute response to any problems. Keep in mind there is a tendency to underestimate the importance of support until there is an emergency and the support is needed. Support is typically a company to company decision that is heavily influenced by the choice of IDS solution.

Challenges inherent in NIDS

Network Architecture Issues

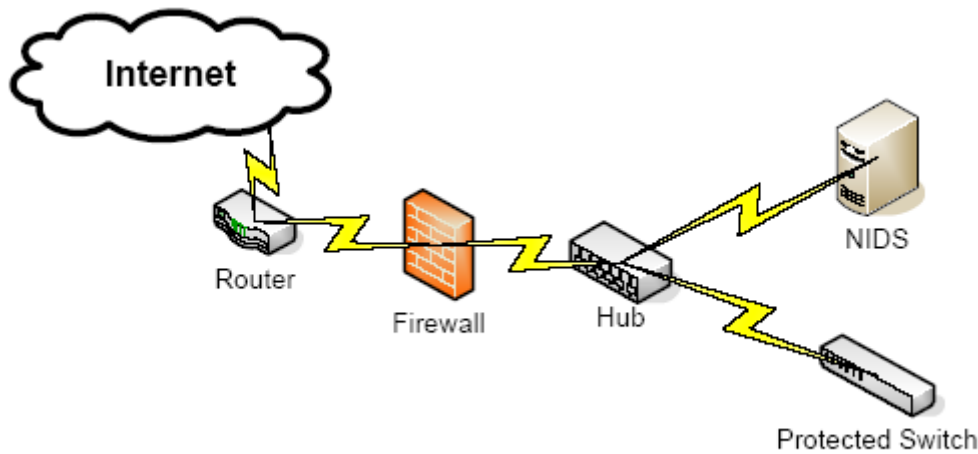
As I have alluded to, there are a number of challenges that must be considered when deciding on a NIDS solution. Some of these challenges are inherent in what an IDS does and other challenges are simply part of the way that a modern network is configured.

First of all, switches are an issue for a NIDS. Recall, the beauty of a switch is that it only forwards traffic to the ports that have devices involved in a given conversation. So if Computer A on port 1 is talking to Computer B on port 2 then only machines connected to ports 1 and 2 will see the traffic. This has many advantages. Switches eliminate collisions, they reduce processing power required on terminating devices and they make malicious packet sniffing much more difficult. This last advantage of switching is a problem for a NIDS. (Laing, 2000)

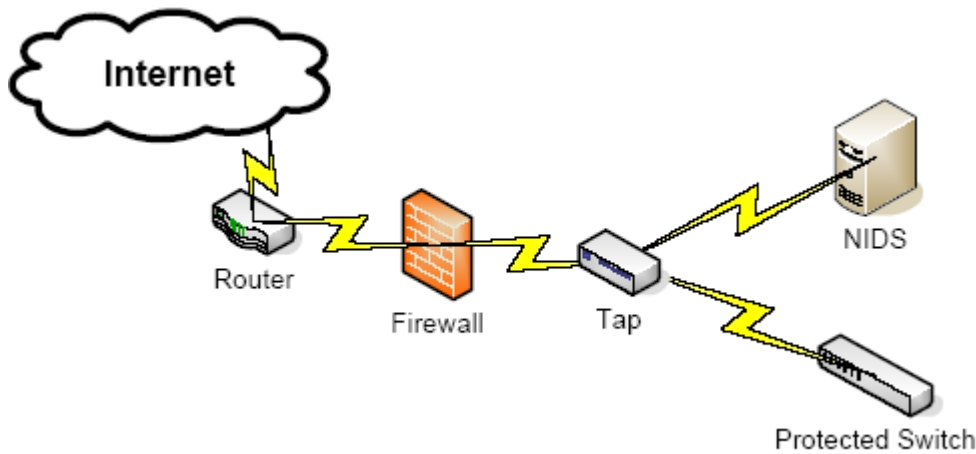
The NIDS needs to be able to look at all of the traffic on a protected network segment. There are a number of ways to achieve this goal all having inherent advantages and disadvantages. The simplest and least efficient method of mitigating this problem is adding a hub inline at a choke point and then plugging the IDS into the hub. This creates a segment that looks something like this: Switch 1 → Hub with attached IDS → Switch 2. Any traffic going between Switch 1 and Switch 2 will be seen by the IDS. (See figure 1.) I say this is the least efficient method because adding a hub breaks many of the advantages of using a switched network. Collisions become an issue again. Full duplex

traffic is eliminated, which cuts your effective network bandwidth in half. The hub is an additional point of failure introduced into the system. (Laing, 2000)

Figure 1



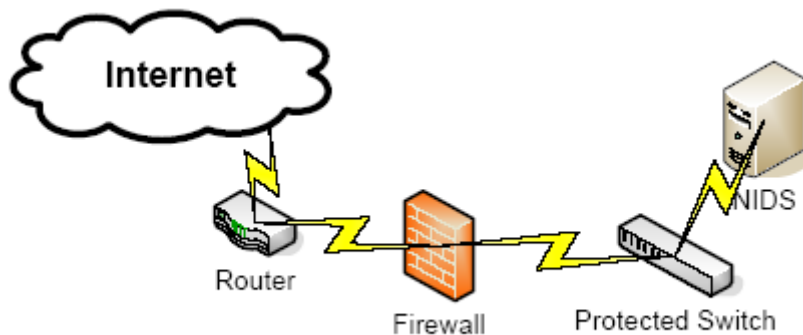
The network tap is another approach to allowing the NIDS to see all the traffic on a switched network. A tap is similar in function to a phone tap. The tap will typically look like 3-port switch. Port 1 will attach to Switch 1, Port 2 will attach to Switch 2 and Port 3 will attach to the NIDS. (See figure 2.) Every packet that is forwarded between Switch 1 and Switch 2 will be mirrored to the NIDS. The tap does not break the full duplex nature of the communications. Most commercial taps can be configured to fail open or closed in the event of a failure of the tap. The major disadvantage of using a tap is the cost. A one port tap can easily cost more than a managed workgroup switch. (Laing, 2000)

Figure 2

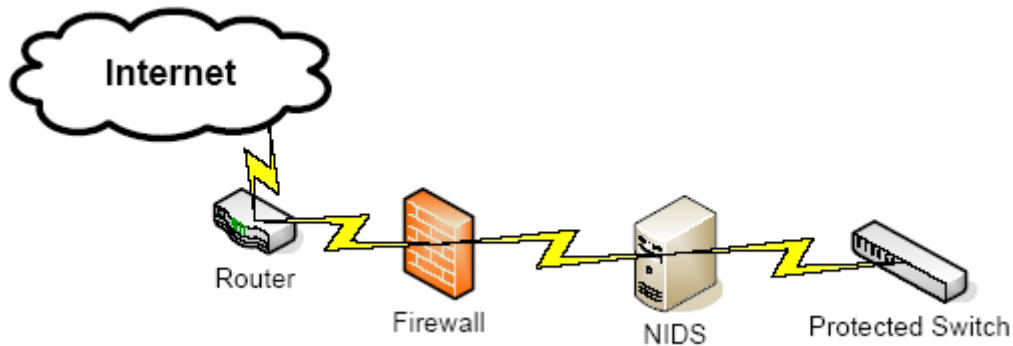
Another popular option for adding a sniffer of any type to a network is the use of a span port on the switch being monitored. A span port is a port that is configured to have a copy of all packets sent to it. (See figure 3.) Much like a tap it does not break the full duplex nature of switched traffic. Buying a switch with a span port can be more expensive than purchasing one without a span port, but most better enterprise switches will have the functionality available without having to upgrade to a more expensive switch. The only place where the cost of a span port should come into place is in a network that is using older switches or inexpensive switches. The major disadvantage of spanning ports is that they can have a detrimental effect on other traffic traversing the switch. If the switch is heavily loaded there may be more traffic going through the switch than can be mirrored onto a single port. Another issue that should be investigated before implementing a spanning port is whether there are any known issues with spanning and the switch. Some implementations of spanning ports have proven to cause large spikes in resource utilization on the switch even at relatively low network utilization levels. One consideration that may or may not be important depending on the location of the NIDS in

the network is that ARP requests are not typically forwarded to the span port so ARP attacks will not be seen. Some NIDS implementations do not have the ability to detect ARP attacks so this may be a moot point depending on the NIDS solutions chosen. One final issue that can come up with spanning ports is that there is usually only one available. If a sniffer is already being used on this port the span port will not be available for the IDS. If the shortcomings of the span port are acceptable it is often the easiest and least expensive way of implementing a NIDS without degrading network speeds. (Laing, 2000)

Figure 3



The final option is an inline NIDS. An inline NIDS looks essentially like a bridge. (See figure 4.) Typically the NIDS will be configured without an IP so that it will not respond to any traffic. The IPS will simply accept traffic on one NIC and pass it back out unchanged on a second NIC like a bridge. This is especially popular with active response IDS systems. In an active response system the NIDS can shut down traffic like a firewall since it is located at a choke point. Most commercial hardware based inline NIDS can fail open or fail closed according to the configuration of the device. Software based inline NIDS typically fail closed, which can be a denial of service issue since a failed IDS will shut off all traffic going to the protected network. (Crothers, 2003)

Figure 4

High Speed Networks

High-speed networks are a problem for NIDS solutions. The speed of the network increases ahead of the processing power available to pull every packet off the wire and process it. Current high-end systems should be able to cope with a marginally loaded gigabit network, but will not be able to come close to coping with a fully loaded 10G network. This is less of an issue for the SMB since smaller businesses do not typically have a need to move to these faster networking speeds as soon as they become available, but this is an issue that must be considered if a company has heavily loaded networks.

There are a number of different approaches that can be taken that will help improve this issue. The first and easiest is separation of the login, and the analysis function onto different machines. This will have the added improvement of being a more secure implementation since a compromise of the logging server will not allow logs to be modified or deleted since the actual logs will be stored on the analysis server.

A second approach if separating the logging from the analysis is not a sufficient solution is to implement multiple logging servers that look at different sections of the network. The analysis system will still be able to take the logs from each logging server

and recombine them back into a consistent overview of the entire network. Given enough logging servers almost any network should be monitorable. (Crothers, 2003)

False Positives and False Negatives

One of the biggest ongoing problem that most IDS implementations have is the number of alerts created. Out of the box, most IDS solutions will have a large proportion of their signatures turned on. This is akin to a default deny approach in firewall management. This means that one of the first tasks that must be undertaken in making a new IDS implementation useful is turning off rules that are not needed. This will help to speed up the processing as well as reduce false positives caused by rules that are irrelevant to the environment at hand. On the other hand the IDS manager must be careful not to turn off rules for services that may be implemented later. There is an argument, which I subscribe to, stating that if you have the computing resources anything that can be monitored should be monitored. The other more surgical approach to reducing false positives is to make bypass rules that allow very specific traffic to skip past a broader rule. (Snort Core Team) As an example, if a particular PC is causing alerts when it connects to a particular server to complete legitimate work a bypass rule can be written telling the NIDS not to look for that PC connecting to that server using that particular troublesome service. Any other attack against the protected service will still trigger an alert. It's a simple solution but it can take a great deal of time since a new rule must be written and tested for each troublesome service. To give an example of just how many alerts can be generated by an untrained IDS system I placed a generic out of the box Snort IDS on a small network of less than 100 machines. In the first two hours Snort generated 27,668 alerts. That consisted of 28 unique alerts. Of the 27,668 alerts a single

machine copying files across the network caused 23,955 of the alerts. It is easy to look at those alerts and throw them out, but in doing so I take the chance that there is a real attack hidden in those 23,955 bogus alerts.

Anomaly detection NIDS are not immune to false positives. Since by definition anomaly detection systems detect anything out of the ordinary there is a high likelihood of getting new false positives any time a new piece of network aware software is introduced to the network. These new pieces of software must be trained into the system.

Then we have the second more worrisome problem, the dreaded false negative. False negatives are attacks that are not recognized by the IDS as what they are. In a pattern matching system this can be caused by a new attack not matching existing rules or new rules not being loaded on a frequent enough basis. This is not to say that anomaly based NIDS systems are immune to false negatives. If an attack looks similar enough to normal traffic it may get past. As a general rule anomaly based systems do have a reputation for being better at detecting unknown attacks. Since anomaly detection looks for anything out of the ordinary there is not a need for the system to know why it is out of the ordinary. It simply triggers an alert. There is the possibility of training attacks into the anomaly based NIDS if there are attack going on during the training period.

Handling Alerts

Alerts are the reason for having an IDS. Without them you simply have a complex sniffer. The problem, as I have already mentioned, is the number of alerts created by the average IDS system. These alerts can be broken down into groups. The first group is the false positive. This will be by far the largest of the groups unless there is an active attack going on. As I mentioned earlier the trick with false positives is to

remove as many of them as possible in a surgical way that does not open the network to false negatives. The second group is legitimate attacks that can be cataloged and ignored. These consist of attacks against systems that are not vulnerable to the attack being tried. An example of an ignorable attack is an IIS attack being attempted against an Apache web server. The system simple is not vulnerable. This type of attack can typically be tuned out so that it does not cause false positives. A second attack that is of minimal concern is an attack against a system that has been patched against the attack being attempted. It is wise to make sure that the attack is not succeeding, but beyond that the attack can simply be cataloged. A final type of attack that some people do not find worth worrying about is reconnaissance attacks such as port scanning. This is an issue that people feel differently about. Many people look at port scanning as an attack while others look at from a more pragmatic approach stating that there are so many people rattling the door if you start trying to go after every one of them you will waste so much time that you will surely miss the real attacks that have a chance of compromising your systems. I belong to the later group, but feel free if you have the time to track down the door rattlers, because they are probably compromising systems somewhere even if it is not your system.

The final group is the alert that the NIDS was implemented for. The positive identification that something is wrong. There should be a procedure in place to handle identification of a potentially successful attack. The response will vary from company to company. I will not try to cover incident response in this short paper but there are numerous good books and papers available on the topic.

Fine Tuning the NIDS

Ideally all systems should be tuned for performance whether they are an IDS or a database server, but the reality is that we often find it more cost effective to simply over spec our servers than to spend the time required to fully tune a system. In *The UNIX Philosophy* Mike Gancarz makes the argument that the time required to make a program run faster is wasted if the program meets today's needs since the hardware of tomorrow will be faster effectively giving you a free speed boost without increasing the actual efficiency of the software. This theory works well in building server and even networks, but this type of thinking will not work when considering tuning a NIDS. The limiting factor for a NIDS is not typically hardware with the possible exception of keeping up with network speeds. The limiting factor is the human IDS manager who must review the copious amounts of data produced by the IDS. Unlike computing power, the human will not double his processing capacity every 18 months. So we must take the time to tune the IDS for efficiency.

This tuning takes a few different forms that I have briefly mentioned before. First it is important to turn off rules that should never be an issue. If you have no Linux machines and no plans to add any Linux systems rules for Linux vulnerabilities can be turned off. You must be careful when turning off rules because there is a tendency for new services and devices to get added unexpectedly. (Snort Core Team) So just because you are not running MS SQL on a server today does not mean you should turn off the rules if you have the available computing resources to leave them in place. As an example, the SQL Slammer worm demonstrated that there are many programs that install MSDE thus creating unknown MS SQL implementations. (CERT/CC, 2003) This is only

one case of programs installing additional potentially vulnerable software. You must also take into consideration what software might get installed as part of the default installation on new clients and server.

The bigger improvements come with reductions of false positives. This improves the efficiency of the human that has to read the alerts that are created by the NIDS. In a signature based IDS this tuning will typically be accomplished by creating filters that allow legitimate traffic that incorrectly triggers an alert to pass while still identifying real attacks. (Crothers, 2003) This is a time consuming task, but in the long run it will save operator time as well as reduce the chances of missing a real attack.

Maintenance

Like any computer system a NIDS will require maintenance. Any maintenance that would routinely be taken on any other system should be taken on a NIDS. In additions since NIDS log a huge amount of data they must regularly have their logs cleaned out. This typically consists of archiving the data in case it is needed later. Some companies may prefer to simply delete old records after they are no longer needed. To a great extend this will depend on the incident response and computer use policies in place.

Active Response IDS aka Intrusion Prevention Systems

How does IPS improve upon NIDS?

A large problem with an IDS system is that for it to be useful it must be continually monitored. Much of the functionality of an IDS is lost if the system is not monitored around the clock. An IDS identifies possible problems and triggers an alert

that a trained IDS administrator can evaluate and use as a basis to decide on a course of action. There can be a large lag time between an alert and action. This is especially true in the case of a SMB where IDS monitoring will be one of many jobs carried out by a small information technology or security staff. Even with these disadvantages IDS is still useful as a logging mechanism in attack investigation, mitigation and recovery.

An IPS looks at the lack of response inherent in IDS and improves on that by allowing the system to automatically make changes based on the alerts that it is creating. These changes can consist of making changes to external firewall or router rules, sending reset packets to stop communications or even blocking the traffic at the IPS. The last approach seems to be the most common IPS solution available today, because it is easier to implement than integration with external system and more effective than sending resets that can be ignored by the attacking computer. (Crothers, 2003)

IPS is the wave of the future for IDS. There are two reasons for this. A traditional IDS is costly to operate because of the ongoing monitoring that is required. Secondly and more importantly IDS solutions are overcoming many of the false positive issues discussed earlier in the paper making automated blocking a possibility. Having said that, IPSs must be very carefully evaluated before implementation in a network.

Additional Challenged and Risks Inherent in IPS

Any system that substantially automates the process of blocking network traffic must be carefully evaluated before implementation. In the case of IDS this is especially true. Consider the issues that are inherent in IDS. The biggest problem with IPS is the false positive. If legitimate traffic begins to trigger rules that are configured to block subsequent traffic from the “attacking” host the network will suddenly start to experience

a denial of service from within. Some IPS systems do have the option of working in IDS only mode until the company is comfortable with the false positive possibilities. Another option in some IPS systems is to allow different behavior for different signatures so extremely urgent attacks can be configured to block while less urgent attacks can be configured to only alert. When considering an IPS, I recommend first implementing the solution in IDS mode and then moving to IPS mode if there is not a significant false positive problem.

On the subject of DOS caused by IPSs it should be pointed out that an attacker can perpetrate a DOS attack by misusing the IPS. If the attacker knows that the IPS is configured to block all subsequent traffic from an attacker's IP address after certain attacks are detected the attacker can begin sending traffic that will trigger this rule from spoofed addresses. This is primarily an issue if the attacker has a small number of addresses that they want to stop from reaching the victim, but given enough resources, such as a bot net, this could be used to cut off very large segments of the Internet in a short amount of time.

Conclusion

For small and medium businesses security is often an afterthought. It is easy for those in charge of these companies to believe that their company is too small to be a target for attack. This is simply not true. Attack on large e-commerce or government sites gets more attention than attacks on SMBs, but that does not mean that SMBs are not being attacked every day of the year.

For years now most companies have realized that a firewall and virus scanners are simply the cost of doing business, and along with other best practices, those are an

excellent first line of defense. In today's more hostile computing environment simple firewalls and anti-virus software are no longer good enough for many businesses. IDS fits in this hole very well. The IDS allows an administrator to watch attacks as they enter the protected network. An IDS will also help in detection of malicious software such as Trojan horses or even spyware that might make in onto systems.

One of the major reasons that IDS has been ignored in all but the largest companies over the years is the cost and complexity of IDS. The complexity is still rather high, but it is becoming less of an issue. A number of less costly commercial solutions and OSS solutions make IDS more affordable than ever for all sizes of business.

IDS is not right for all companies, but it has evolved enough that companies that have considered implementing IDS in the past and decided it was not a viable option should reconsider now. For companies that have not evaluated IDS in the past now is an excellent time to evaluate it for the first time. Most companies will be surprised at the number of attacks that are making it into their network. These may be ineffectual attacks that can be ignored, but it is better to know that the attacks are happening than to blindly assume security. When the attack is potentially successful it is better to find out proactively rather than after substantial damage has been done.

References

- Bace, Rebecca Gurley. (2000). *Intrusion detection*. Indianapolis, IN: Macmillan Technical Publishing.
- Crothers, Tim. (2003). *Implementing intrusion detection systems: A hands-on guide for securing the network*. Indianapolis, IN: Wiley Publishing, Inc.
- CERT/CC. (January 27, 2003) CERT® Advisory CA-2003-04 MS-SQL Server Worm. Retrieved October 18, 2005 from <http://www.cert.org/advisories/CA-2003-04.html>.
- Graham, Robert. (October 16, 2002). FAQ: Network intrusion detection systems. Retrieved May 22, 2005 from http://www.seconf.net/intrusion_detection/FAQ_Network_Intrusion_Detection_Systems_.html
- Gancarz, Mike. (1995). *The UNIX Philosophy*. Newton, MA: Butterworth-Heinemann.
- Laing, Brian. (2000). How to guide – Implementing a network based intrusion detection system. Retrieved May 22, 2005 from <http://www.snort.org/docs/iss-placement.pdf>
- Snort Core Team. (n.d.) The Snort FAQ. Retrieved May 22, 2005 from <http://www.snort.org/docs/faq/1Q05/faq.pdf>.