Global Information Assurance Certification Paper

Host- vs. Network-Based Intrusion Detection Systems

Introduction

Within the last five years or so, organizations have come to incorporate information technology into their internal operations and business solutions on an enormous scale [3]. This phenomenon is complimented by the increasing need for remote access to system resources due to the growing trend toward telecommuting and the increased utilization of video and voice conferencing. In addition, many local and federal government functions are now conducted over the Internet. As a result, both business and government have become critically dependent on both internal and external computer networks. In many respects, this is an encouraging and positive condition; these networks allow for a more efficient workplace, a more versatile and mobile workforce, and facilitate such things as global communication and electronic commerce.

However, in some ways, this leaves the businesses and government organizations in a dangerous position. Crime, for example, that would traditionally be directed at a specific outlet of a store or a strategic federal office, will now likely be directed at the information systems maintained by these bodies. Since these organizations are so dependent on network operation and connectivity, most with mission-critical resources residing on these networks, they leave themselves extremely susceptible to malicious activity that is directed at their networks. Rightly so, awareness about security measures for these systems has increased immensely. It is common for a company to implement a firewall or a security policy, but experience has shown these to be dramatically insufficient [14].

Both industry and government will come to depend on more advanced and integrated security measures to protect their systems from attacks. Though several methods exist for providing network security, arguably the best tool for doing this is the use of intrusion detection systems, these systems are the logical complement of network firewalls and security management [1]. Intrusion detection systems are available in two flavors, host-based and network-based. This paper will first explain what intrusion detection is, then explain and evaluate the two approaches to intrusion detection systems individually, and finally analyze the converging trends of these two methods as well as touch on the evolution of intrusion detection systems. It should be noted that this text is not intended to be a survey or comparison of current intrusion detection systems, for those interested, a partial listing of these systems is available on the Internet [15].

Foundation

Intrusion detection systems are security systems that collect information from various types of system and network sources, and analyzes this data in an attempt to detect activity that may constitute an attack or intrusion on the system. This data also helps computer systems and systems administrators prepare for and deal with attacks, or intrusion attempts, directed at their networks [1], [2]. In addition, the features of an intrusion detection system lets system managers to more easily handle the monitoring,

© SANS Institute 2000 - 2005 Author retains full rights.

audit, and assessment of their systems and networks, which is a necessary part of security management [1]. This monitoring process is an ongoing one, as the intrusion detection system must change as the types of attacks change.

As will be seen, even though the monitoring techniques and targets differ, all of these systems provide sentinel functions, which will send alarms and alerts to the responsible parties when activities of interest occur on the network. In some cases, these systems will allow users to define real-time responses to attacks [2]. For several years, there has been a continuing debate on whether host- or network-based systems are the superior strategy. In the following sections, the principles of the two approaches will be presented individually so their differences will be clear.

Host-based Systems

Host-based intrusion detection systems are aimed at collecting information about activity on a particular single system, or host [1]. These host-based agents, which are sometimes referred to as sensors, would typically be installed on a machine that is deemed to be susceptible to possible attacks. The term "host" refers to an individual computer, thus a separate sensor would be needed for every machine. Sensors work by collecting data about events taking place on the system being monitored. This data is recorded by operating system mechanisms called audit trails [1], [2], [11], [14]. Other sources from which a host-based sensor can obtain data, "include system logs, other logs generated by operating system processes, and contents of objects not reflected in standard operating system audit and logging mechanisms" [1]. These logs are for the most part simple text files, which are written a few lines at a time, as events occur and operations on a system take place.

As host-based systems rely heavily on audit trails, they become limited by these audit trails, which are not provided by the manufacturers who design the intrusion detection system itself. As a result, theses trails may not necessarily support the needs of the intrusion detection system, leading some to conclude that having more effective host-based systems, "may require the developer to amend the operating system kernel code to generate event information. This approach extracts a cost in performance, which might be unacceptable for customers running computationally greedy applications" [2], [13].

Despite this limitation, audit trails are still considered to be the source of choice for host-based intrusion detection information. This continues to be true, first, because of the existing aim of operating systems at protecting its audit layer; and second, for the level of detail that audit trails provide [2]. Clearly, considering the objective of intrusion detection systems, the detail provided is particularly important in analyzing patterns of attack. More importantly, "[the] information allows the intrusion detection system to spot subtle patterns of misuse that would not be visible at a higher level of abstraction" [2]. The fact that audit trails are protected by the operating systems itself offers some assurance that audit trails have not been improperly modified.

The information collected through audit trails can arm the host-based sensor with useful data about the system and its users. For example, audit trails may contain information about subjects responsible for an event, as well as any objects related to that

event. The host-based sensor can recover which process initiated an event, and the current and original user identifications associated with that event, in case the user identification changes [2]. These pieces of data can be crucial in determining from what program and by what user a potential network attack originated, which will obviously help in stopping future attacks. However, in the case of an attack from within, this may also be useful in determining culpability in order to pursue punitive measures against the user.

As useful as the data is, a common criticism of host-based systems lies with the amount of data they can offer. The configuration of the sensors must obviously collect detailed enough information to identify abnormalities on a host, so the more refined the data captured, the better the sensor should work. The problem is that, as the sensors gather finer levels of detail, they accumulate large amounts of data that take up significant storage [1], [13]. In addition, because, "both the volume and complexity of the data rise with greater detail … it makes it difficult for an adversary to circumvent the audit process entirely, the greater volume and complexity of the data make it easier in practice for intruders to hide their footprints" [2]. This sort of irony becomes the burden that designers and analysts must overcome so that host-based sensors avoid becoming cumbersome, while remaining effective.

Host-based intrusion detection systems are desirable for several reasons. As briefly mentioned above, because host-based systems can monitor access to information in terms of "who accessed what," these systems can trace malicious or improper activities to a specific user ID [1], [9]. This is always important as it can identify whether a person inside the organization is responsible for the improper use of company resources, for example, if a person's desk computer is being used to launch network attacks. The problem then is to determine if that employee at any time had knowledge of the illicit events. Host-based sensors are also useful in that they can keep track of the behavior of individual users [1]. This can help catch attacks while they are happening or possibly stop a potential attack before it affect the system. If a pattern is observed that is similar to past attacks or that is suggestive of an attack, activity to and from that workstation can be stopped, foiling the attack. This ability can be an especially useful in systems in which remote access to system resources is common.

Host-based systems are valuable in that they are, in some ways, very versatile. They have the ability to operate in environments that are encrypted, as well as over a switched network topology [1]. Also, since host-based systems are necessarily disbursed throughout a system, there are certain cost advantages associated with them. "[Host-based] systems can distribute the load associated with monitoring across available hosts on large networks, thereby cutting deployment costs" [1]. The distribution of host-based systems also allows them to be scalable [14], the load is spread evenly over a network which is a valuable asset when network traffic becomes very large. Although this does offer a margin of cost reduction, both in terms of money and network performance, the discussion below will give a clearer perspective of on the costs involved with host-based intrusion detection systems.

Host-based systems also have several disadvantages. One observation is that they cannot see network traffic [1]. Since they are not designed to see network traffic, but to run on a single system, it seems unfair to characterize this as a negative point. No matter

how it is viewed, this is an inherent limitation of host-based systems. As was explained above, host-based systems are heavily dependant on host operating system. Any existing vulnerabilities to this system will weaken the integrity of the host-based sensor [1]. If an intruder can find and exploit one of these weaknesses, this could lead to an attack which is hard to catch and a vulnerability which is difficult to correct.

The problem of system resources was explained above, since audit trails are used as the source of information, they can be very costly, taking up significant storage space as well as increase hosting server load. There are also large costs in setting up a host-based system. Again, since individual sensors are required for each host, "management and deployment costs associated with host-based systems are usually greater than in other approaches" [1]. Accordingly, in very large environments, a host-based approach could be economically infeasible [14].

Lastly, host-based intrusion detection systems have the chronic problem of portability. The sensors are host-based, so they have to be compatible with the platform they are running over [1]. This lack of cross-platform support would represent a major obstacle for a corporation wishing to employ a host-based solution. Although more products are supporting a broader range of platforms, an interested company's operating system may not be in the list [14].

Network-based Systems

Network-based intrusion detection systems offer a different approach. "These systems collect information from the network itself," [1] rather than from each separate host. They operate essentially based on a "wiretapping concept," information is collected from the network traffic stream, as data travels on the network segment [2], [8], [14]. The intrusion detection system checks for attacks or irregular behavior by inspecting the contents and header information of all the packets moving across the network. The network sensors come equipped with "attack signatures" that are rules on what will constitute an attack [7], [14], and most network-based systems allow advanced users to define their own signatures. This offers a way to customize the sensors based on an individual network's needs and types of usage. The sensors then compare these signatures to the traffic that they capture, this method is also known as packet sniffing [1], [14], and allows the sensor to identify hostile traffic.

Using network data as a primary source if information is desirable in several ways. To start, running network monitors does not degrade the performance of other programs running over the network. This low performance cost is due to the fact that the monitors only read each packet as they come across its network segment [1], [2]. The operation of the monitors will be transparent to system users [14], and this is also significant for the intrusion detection system itself. The transparency of the monitors, "decreases the likelihood that an adversary will be able to locate it and nullify its capabilities without significant effort" [2]. This decreased vulnerability strengthens the intrusion detection system, and adds another measure of security. From a financial perspective, network-based systems are very desirable. The primary resource for these monitors is storage space, so companies could use older and slower equipment to do this work [2], rather

than purchase additional equipment. This could significantly save on deployment costs.

Network-based systems are also extremely portable. They only monitor traffic over a specific network segment, and are independent of the operating systems that they are installed on. "Deployed network-based intrusion detection sensors will listen for all attacks, regardless of the destination operating system type" [14]. This offers more options for businesses that run specialized software or software they have developed in-house, which will become increasingly attractive as the newer UNIX-based operating systems continue to increase in popularity. Adding to their convenience, network-based sensors can be inserted easily on part of a network and data can be collected with minimal work. In many cases, all that is required to collect information for analysis is the configuration of a network card [1]. This is beneficial in situations where network topology changes or where system resources have been moved, the intrusion detection system monitors can be moved and used as needed.

However, network-based solutions have their share of problems. As discussed earlier, the sensors spot attacks based on their attack signatures. These signatures are written based on data collected from known and previous attacks, and this unfortunately ensures that these signatures "will always be a step behind the latest underground exploits" [14]. What is worse is that, although intrusion detection system vendors offer regular updates to their signature databases, many have not caught up in defining signatures for all known attacks [14]. While these systems can still prevent many attacks, serious coordinated attacks—the kind for which no signatures have been predefined--have the potential to do the most damage.

The second major issue with network-based intrusion detection approaches is scalability. Network monitors must inspect every packet that is passed through the segment they are placed on. It has been demonstrated that network-based systems have difficulty keeping up on 100 Mbps environments [14], they simply can't handle it, and now the trend is moving toward gigabit speeds. As these high-speed networks become more common, intruders will be able to identify them, and they will no doubt be targeted with attacks gauged at specifically exploiting this weakness. Strategic placement of network sensors can help to alleviate this, but systems with heavy traffic will still encounter this problem.

Encryption and switching represent two further limitations of network-based approaches. First, if network traffic is encrypted, an agent cannot scan the protocols or the content of these packets [1], [7]. Second, the nature of switches makes network monitoring extremely difficult. "[I]n the case of switched networks the network switch acts to isolate network connections between hosts so that a host can only see the traffic that is addressed to it" [2]. In these cases, a network-based monitor is essentially reduced to monitoring a single host, defeating much of the intent of the monitor. Some switches can now support a port for monitoring and scanning, which offers a partial solution to this problem [1].

In addition, network monitors are unable to see traffic travelling on other communication media, such as dial-up phone lines [2]. This is an increasing concern as organizations employ a greater number of telecommuters, since their traffic cannot be monitored using this approach. This problem is actually part of a larger issue. The network sensors have a degree of blindness to host activity. "Although some network-

based systems can infer from network traffic what is happening on hosts, they cannot tell the outcomes of commands executed on the host. This is an issue in detection, when distinguishing between user error and malfeasance" [1]. This limitation could lead to numerous false-positives, which is an undesirable situation where an intrusion detection system falsely identifies something as an attack. Intrusion detection systems are configured and signatures are carefully written to minimize the instances of false-positives.

A Superior Method?

In the sections above, this paper has made an attempt to present each approach to intrusion detection systems, explaining the two types and outlining their strengths and weaknesses, without making a comparison. Though they both have the same goal, the two approach this goal in very different ways. Also, the types of systems are designed to look for separate classifications of things. Therefore, holding the two side by side, evaluating them in hopes of determining a winner, is inappropriate. The host-based systems do offer an approach that scales better, but implementing this type of intrusion detection system requires a high degree of expertise about the operating system that the sensors will run on [6]. Also, the lack of cross-platform support is a considerable problem [14]. On the other hand, network-based solutions are more portable [14], and are easier to implement [1], but have the growing problem that they cannot keep up with heavy traffic or with high network speeds.

From an attack perspective, the situation is similar. Network-based intrusion detection systems are appealing because of the way they inspect traffic, "[these] network monitors can see evidence of certain classes of traffic that are not visible to host-based systems." Attacks from malformed or "crafted" packets, packet storms, and many denial of service attacks can only be discovered with sensors on a network [2], [6]. Host-based systems, however, offer the counter argument. An attacker attempting to infiltrate a host system may do so through a dial-up connection, which cannot be seen by network monitors, only by a sensor on the target host. Further, only host-based sensors can examine the results of commands that are executed on a host system, which could possibly be malicious or simply against a security policy. In many ways, neither method offers a complete intrusion detection solution.

The latest arguments suggest that the best solution is one that will incorporate both methods [1], [5], [6], [7], [14]. A system that integrates both host- and network-based characteristics seems intuitively the most logical approach. So, one may wonder why it has been only recently that host- and network-based methods have started to become integrated. Why didn't vendors of intrusion detection systems just initially begin with a design that took both aspects into consideration?

The explanation is quite frankly a question of the security needs of computer systems over time. The first intrusion detection systems designed were only run on a single host. When the need for a tool to monitor improper activity became evident, the systems in question were single mainframe computers, with the intrusion detection tool as well as the users local to that computer [5]. The mainframe's audit information would be

analyzed and suspicious events reported, and outside interactions with the mainframe were generally very rare [5]. Clearly, intrusion detection was a much more simplified field compared to what it is today.

Intrusion detection started to become more complex as mainframe environments were being replaced by distributed systems. "In a distributed environment, users hop from one machine to another, possibly changing their identities during their moves and launching their attacks on several systems" [5]. Something had to be done to handle this new forma of attack, enabled by the advent of these computer networks. Research focused on extending the host-based concept to small groups of workstations, which would require several single host intrusion detection tools to communicate with each other [5], [16]. As these small networks became more complex, intrusion detection monitors required more efficient communication, not only between the monitors themselves, but also between the workstation operating system and the monitors.

Unfortunately, simply extending host-based intrusion detection to networks was not acceptable, considering such heavily interconnected environments [16]. As most networks moved onto the Internet, they became open to a different array of attacks. These new attacks, such as DNS spoofing, TCP hijacking, and ping of death attacks, focused on attacking the network itself instead of on a single machine [4], [5]. These attacks were facilitated by the widespread use of the Internet and the need for communication between several networks, and forced intrusion detection systems to focus on attacks to the network itself. Thus, the focus in intrusion detection shifted to examining network traffic in order to determining if an attack was taking place.

So, although an integrated approach does seem to be preferable, it was not always the case, since the scope of computer security has grown so noticeably. Some vendors are working to expand their products to produce integrated solutions, but this is taking place very slowly. From a marketing viewpoint, a vendor can make more money selling a network-based and a host-based intrusion detection system to its customers, rather than integrate the approach and only have a single product to sell [6].

Another problem for both types of intrusion detection systems is the lack of a uniform terminology across different vendors. "For example, if a 'Winnuke' attack is executed on a helpless Microsoft Windows 95 machine, some intrusion detection systems may identify this as an 'Out of Band Windows Attack,' while others might call it a 'NetBIOS OOB attack,' and still others might just say 'Winnuke,' or 'Winuke'" [14]. This is a problem not only for organizations attempting to implement different systems, but also makes it difficult for computer security professionals to become more educated and knowledgeable about the field. This problem, however, is currently being addressed in the Common Vulnerability and Exposure (CVE) project [10], [14]. With strong vendor support, this initiative would help to enhance communication in the both the fields of intrusion detection and vulnerability assessment.

Conclusions

Though the increasing need for computer security in both public and private domains seems obvious, it is a subject which is too often addressed casually, enforced

selectively—with exceptions being made to numerous users at varying levels, and given top priority only after the fact. Intrusion detection systems represent a crucial component of an effective computer security system. This work has been an attempt to define and analyze both host- and network-based systems, outlining how the two approached work and identifying their respective strengths and deficiencies. The goal was not to compare the two in terms of which is superior, but to quantify the type of intrusions each looks for and the realm of security each of the approaches was intended to provide for. The two methods are converging into one integrated system, but this goal has not yet been achieved. Until this becomes a reality, many are recommending the use of both host- and network-based systems [1], [6], [14].

Though this paper focused on intrusion detection systems, one should not draw from this that intrusion detection systems alone will ensure the security of a computer network, nor that simply installing these systems will be an effective means to thwart would be intruders. Intrusion detection systems, powerful as they can be, represent only one of the available tools to provide system security. They are certainly necessary, but by no means sufficient. Intrusion detection systems must be complimented by not only such things are firewalls, vulnerability assessment, and a comprehensive security policy. Organizations must also have systems and security personnel who are experienced and extremely knowledgeable about the intrusion detection systems themselves and the environments in which they are running [12]. Finally, one should never wander far from the principle that, no matter how comprehensive the security tools implemented are, a systems is never impenetrable. A totally secure network or computer system is a paradigm, and can only be viewed as a kind of 'asymptote,' one which we can merely approach from many different directions, but never actually reach.

References:

[1] Bace, Rebecca: *An Introduction to Intrusion Detection & Assessment*. Infidel Inc., prepared for ICSA Inc. Copyright 1998.

[2] Bace, Rebecca Gurley: *Intrusion Detection*. Copyright 2000 by Macmillan Technical Publishing, ISBN 1-57870-185-6

[3] Brackney, R: *Cyber-Intrusion Response*. Reliable Distributed Systems, 1998. Proceedings. Seventeenth IEEE Symposium on, 1998, Page(s): 413 –415.

[4] Chang, H.Y et al: *DecIdUouS: Decentralized Source Identification for Network-Based Intrusions*. Proceedings of the Sixth IFIP/IEEE International Symposium on, 1999, Page(s): 701 -714

[5] Debar, H; Dacier, M; Wespi, A: *Towards a Taxonomy of Intrusion Detection Systems*. Computer Networks, vol 31, 1999, Page(s): 805-822.

[6] Hawthorn, E. K: Intrusion Detection. Lecture presented to INFS 762 at George Mason University, November 13, 2000.

[7] Higgins, Kelly Jackson: *A Welcome Intrusion*. Internet Week Magazine, May 23, 2000, http://www.internetwk.com/lead/lead052300.htm.

[8] Kaeo, M: *Designing Network Security*, by Cisco Systems. Copyright 1999 by Macmillian Technical Publishing, ISBN 1-57870-043-4.

[9] Lunt, T.F; Jagannathan, R; Lee, R; Whitehurst, A; Listgarten, S: *Knowledge-Based Intrusion Detection*. AI Systems in Government Conference, 1989, Proceedings of the Annual, 1989, Page(s): 102 –107.

[10] Mann D. E; Christey, S. M: *Towards a Common Enumeration of Vulnerabilities*. Copyright 1999 by The MITRE Corporation, (damann, coley)@mitre.org.

[11] Mounji; Le Charlier, B: *Continuous Assessment of a Unix Configuration: Integrating Intrusion Detection and Configuration Analysis*. Network and Distributed System Security, 1997. Proceedings, 1997, Page(s): 27 –35A.

[12] Northcut, S: *Network Intrusion Detection, an Analyst's Handbook*. Copyright 1999 by New Riders Publishing, ISBN 0-7357-0868-1.

[13] Reilly, M; Stillman, M: *Open Infrastructure for Scalable Intrusion Detection*.

Information Technology Conference, 1998. IEEE, 1998, Page(s): 129 –133

[14] Shipley, Greg : *Intrusion Detection, Take Two*. Network Computing Magazine, November 15, 1999, http://www.networkcomputing.com/1023/1023f1.html.

[15] Sobirey, M: Intrusion Detection Systems Bibliography. Internet: http://www-rnks.informatik.tu-cottbus.de/~sobirey/ids.html.

[16] White, G.B.; Huson, M.L: *Cooperating Security Managers: A Peer-Based Intrusion Detection System*. MILCOM '96, Conference Proceedings, IEEE Volume: 2, 1996, Page(s): 468 -472 vol.2.