Intrusion Detection Systems

Rebecca Bace[1] and Peter Mell[2]

[1] Infidel, Inc., Scotts Valley, CA

[2] National Institute of Standards and Technology

Intrusion Detection Systems

Rebecca Bace[3], Peter Mell[4]

1. Introduction

Intrusion detection systems (IDSs) are software or hardware systems that automate the process of monitoring the events occurring in a computer system or network, analyzing them for signs of security problems. As network attacks have increased in number and severity over the past few years, intrusion detection systems have become a necessary addition to the security infrastructure of most organizations. This guidance document is intended as a primer in intrusion detection, developed for those who need to understand what security goals intrusion detection mechanisms serve, how to select and configure intrusion detection systems for their specific system and network environments, how to manage the output of intrusion detection systems, and how to integrate intrusion detection functions with the rest of the organizational security infrastructure. References to other information sources are also provided for the reader who requires specialized or more detailed advice on specific intrusion detection issues.

2. Overview of Intrusion Detection Systems

2.1. What is intrusion detection?

Intrusion detection is the process of monitoring the events occurring in a computer system or network and analyzing them for signs of *intrusions*, defined as attempts to compromise the confidentiality, integrity, availability, or to bypass the security mechanisms of a computer or network. Intrusions are caused by attackers accessing the systems from the Internet, authorized users of the systems who attempt to gain additional privileges for which they are not authorized, and authorized users who misuse the privileges given them. Intrusion Detection Systems (IDSs) are software or hardware products that automate this monitoring and analysis process.

2.2. Why should I use Intrusion Detection Systems?

Intrusion detection allows organizations to protect their systems from the threats that come with increasing network connectivity and reliance on information systems. Given the level and nature of modern network security threats, the question for security professionals should not be *whether* to use intrusion detection, but *which* intrusion detection features and capabilities to use.

IDSs have gained acceptance as a necessary addition to every organization’s security infrastructure. Despite the documented contributions intrusion detection technologies make to system security, in many organizations one must still justify the acquisition of IDSs. There are several compelling reasons to acquire and use IDSs:

1. To prevent problem behaviors by increasing the perceived risk of discovery and punishment for those who would attack or otherwise abuse the system,

[3] Infidel, Inc., Scotts Valley, CA

[4] National Institute of Standards and Technology

2. To detect attacks and other security violations that are not prevented by other security measures,
3. To detect and deal with the preambles to attacks (commonly experienced as network probes and other "doorknob rattling" activities),
4. To document the existing threat to an organization
5. To act as quality control for security design and administration, especially of large and complex enterprises
6. To provide useful information about intrusions that do take place, allowing improved diagnosis, recovery, and correction of causative factors.

2.2.1. Preventing problems by increasing the perceived risk of discovery and punishment of attackers

A fundamental goal of computer security management is to affect the behavior of individual users in a way that protects information systems from security problems. Intrusion detection systems help organizations accomplish this goal by increasing the perceived risk of discovery and punishment of attackers. This serves as a significant deterrent to those who would violate security policy.

2.2.2. Detecting problems that are not prevented by other security measures

Attackers, using widely publicized techniques, can gain unauthorized access to many, if not most systems, especially those connected to public networks. This often happens when known vulnerabilities in the systems are not corrected.

Although vendors and administrators are encouraged to address vulnerabilities (e.g. through public services such as ICAT, http://icat.nist.gov) lest they enable attacks, there are many situations in which this is not possible:

- In many legacy systems, the operating systems cannot be patched or updated.
- Even in systems in which patches can be applied, administrators sometimes have neither sufficient time nor resource to track and install all the necessary patches. This is a common problem, especially in environments that include a large number of hosts or a wide range of different hardware or software environments.
- Users can have compelling operational requirements for network services and protocols that are known to be vulnerable to attack.
- Both users and administrators make errors in configuring and using systems.
- In configuring system access control mechanisms to reflect an organization's procedural computer use policy, discrepancies almost always occur. These disparities allow legitimate users to perform actions that are ill advised or that overstep their authorization.

In an ideal world, commercial software vendors would minimize vulnerabilities in their products, and user organizations would correct all reported vulnerabilities quickly and reliably. However, in the real world, this seldom happens thanks to our reliance on commercial software where new flaws and vulnerabilities are discovered on a daily basis.

Given this state of affairs, intrusion detection can represent an excellent approach to protecting a system. An IDS can detect when an attacker has penetrated a system by exploiting an uncorrected or uncorrectable flaw. Furthermore, it can serve an important function in system protection, by bringing the fact that the system has been attacked to the attention of the administrators who can contain and recover any damage that results. This is far preferable to simply ignoring network security threats where one allows the attackers continued access to systems and the information on them.

2.2.3. Detecting the preambles to attacks (often experienced as network probes and other tests for existing vulnerabilities)

When adversaries attack a system, they typically do so in predictable stages. The first stage of an attack is usually probing or examining a system or network, searching for an optimal point of entry. In systems with no IDS, the attacker is free to thoroughly examine the system with little risk of discovery or retribution. Given this unfettered access, a determined attacker will eventually find a vulnerability in such a network and exploit it to gain entry to various systems.

The same network with an IDS monitoring its operations presents a much more formidable challenge to that attacker. Although the attacker may probe the network for weaknesses, the IDS will observe the probes, will identify them as suspicious, may actively block the attacker's access to the target system, and will alert security personnel who can then take appropriate actions to block subsequent access by the attacker. Even the presence of a reaction to the attacker's probing of the network will elevate the level of risk the attacker perceives, discouraging further attempts to target the network.

2.2.4. Documenting the existing threat

When you are drawing up a budget for network security, it often helps to substantiate claims that the network is likely to be attacked or is even currently under attack. Furthermore, understanding the frequency and characteristics of attacks allows you to understand what security measures are appropriate to protect the network against those attacks.

IDSs verify, itemize, and characterize the threat from both outside and inside your organization's network, assisting you in making sound decisions regarding your allocation of computer security resources. Using IDSs in this manner is important, as many people mistakenly deny that anyone (outsider or insider) would be interested in breaking into their networks. Furthermore, the information that IDSs give you regarding the source and nature of attacks allows you to make decisions regarding security strategy driven by demonstrated need, not guesswork or folklore.

2.2.5. Quality control for security design and administration

When IDSs run over a period of time, patterns of system usage and detected problems can become apparent. These can highlight flaws in the design and management of security for the system, in a fashion that supports security management correcting those deficiencies before they cause an incident.

2.2.6. Providing useful information about actual intrusions

Even when IDSs are not able to block attacks, they can still collect relevant, detailed, and trustworthy information about the attack that supports incident handling and recovery efforts. Furthermore, this information can, under certain circumstances, enable and support criminal or civil legal remedies. Ultimately, such information can identify problem areas in the organization's security configuration or policy.

2.3. Major types of IDSs

There are several types of IDSs available today, characterized by different monitoring and analysis approaches. Each approach has distinct advantages and disadvantages. Furthermore, all approaches can be described in terms of a generic process model for IDSs.

2.3.1. Process model for Intrusion Detection

Many IDSs can be described in terms of three fundamental functional components:

- *Information Sources* – the different sources of event information used to determine whether an intrusion has taken place. These sources can be drawn from different levels of the system, with network, host, and application monitoring most common.
- *Analysis* – the part of intrusion detection systems that actually organizes and makes sense of the events derived from the information sources, deciding when those events indicate that intrusions are occurring or have already taken place. The most common analysis approaches are *misuse detection* and *anomaly detection*.
- *Response* – the set of actions that the system takes once it detects intrusions. These are typically grouped into active and passive measures, with active measures involving some automated intervention on the part of the system, and passive measures involving reporting IDS findings to humans, who are then expected to take action based on those reports.

2.3.2. How do I distinguish between different Intrusion Detection approaches?

There are several design approaches used in Intrusion Detection. These drive the features provided by a specific IDS and determine the detection capabilities for that system. For those who must evaluate different IDS candidates for a given system environment, these approaches can help them determine what goals are best addressed by each IDS.

2.3.3. Architecture

The architecture of an IDS refers to how the functional components of the IDS are arranged with respect to each other. The primary architectural components are the Host, the system on which the IDS software runs, and the Target, the system that the IDS is monitoring for problems.

2.3.3.1. Host-Target Co-location

In early days of IDSs, most IDSs ran on the systems they protected. This was due to the fact that most systems were mainframe systems, and the cost of computers made a separate IDS system a costly extravagance. This presented a problem from a security point of view, as any attacker that successfully attacked the target system could simply disable the IDS as an integral portion of the attack.

2.3.3.2. Host-Target Separation

With the advent of workstations and personal computers, most IDS architects moved towards running the IDS control and analysis systems on a separate system, hence separating the IDS host and target systems. This improved the security of the IDS as this made it much easier to hide the existence of the IDS from attackers.

2.3.4. Goals

Although there are many goals associated with security mechanisms in general, there are two overarching goals usually stated for intrusion detection systems.

2.3.4.1. Accountability

Accountability is the capability to link a given activity or event back to the party responsible for initiating it. This is essential in cases where one wishes to bring criminal charges against an attacker. The goal statement associated with accountability is: *"I can deal with security attacks that occur on my systems as long as I know who did it (and where to find them.)"* Accountability is difficult in TCP/IP networks, where the protocols allow attackers to forge the identity of source addresses or other source identifiers. It is also extremely difficult to enforce accountability in any system that employs weak identification and authentication mechanisms.

2.3.4.2. Response

Response is the capability to recognize a given activity or event as an attack and then taking action to block or otherwise affect its ultimate goal. The goal statement associated with response is *"I don't care who attacks my system as long as I can recognize that the attack is taking place and block it."* Note that the requirements of detection are quite different for response than for accountability.

2.3.5. Control Strategy

Control Strategy describes how the elements of an IDS is controlled, and furthermore, how the input and output of the IDS is managed.

2.3.5.1. Centralized (Figure 1)

Under centralized control strategies, all monitoring, detection and reporting is controlled directly from a central location

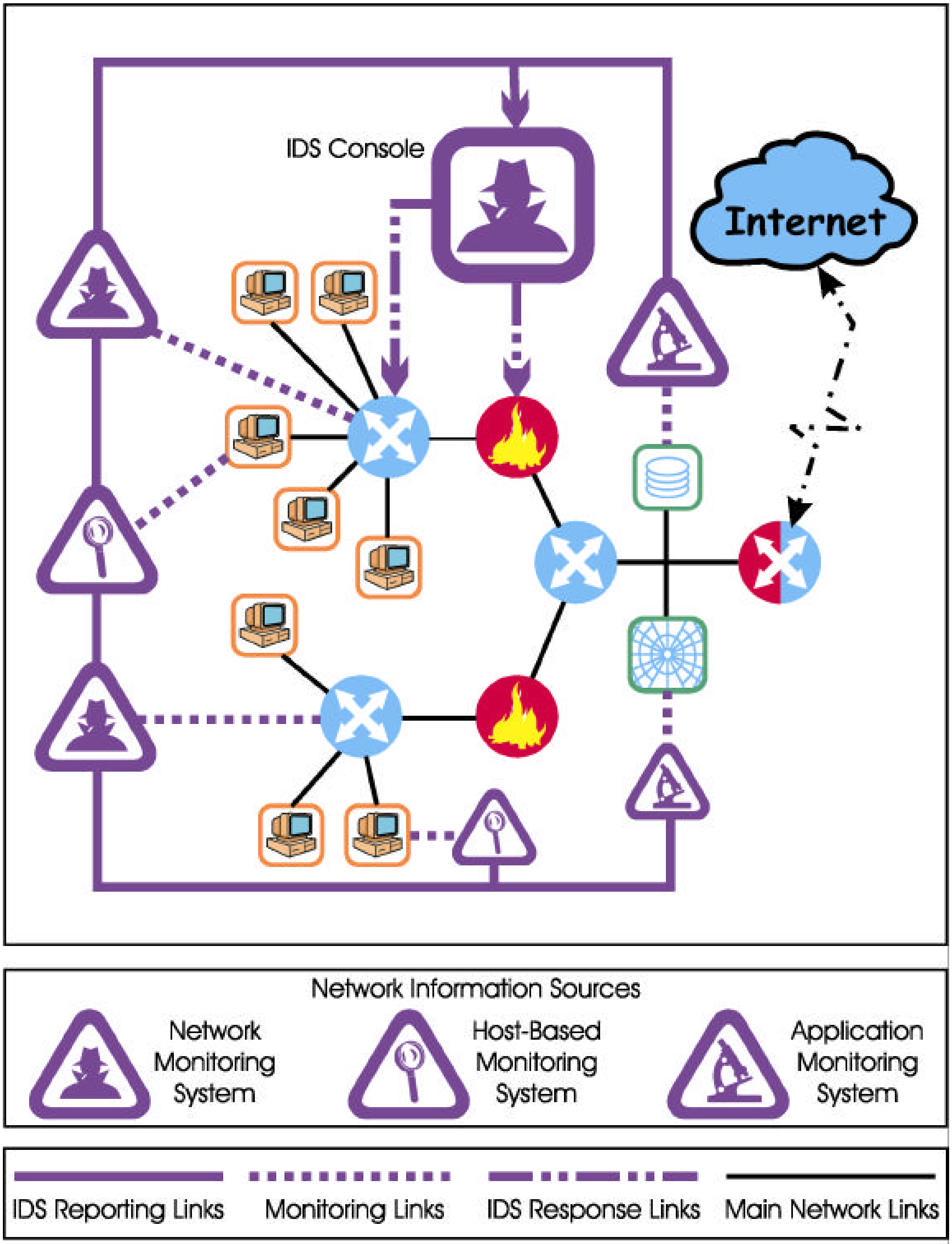
IDS Console
Internet
Network Information Sources
Network Monitoring System
Host-Based Monitoring System
Application Monitoring System
IDS Reporting Links
Monitoring Links
IDS Response Links
Main Network Links

Figure 1: Centralized Control

2.3.5.2. Partially Distributed (Figure 2)

Monitoring and detection is controlled from a local control node, with hierarchical reporting to one or more central location(s).

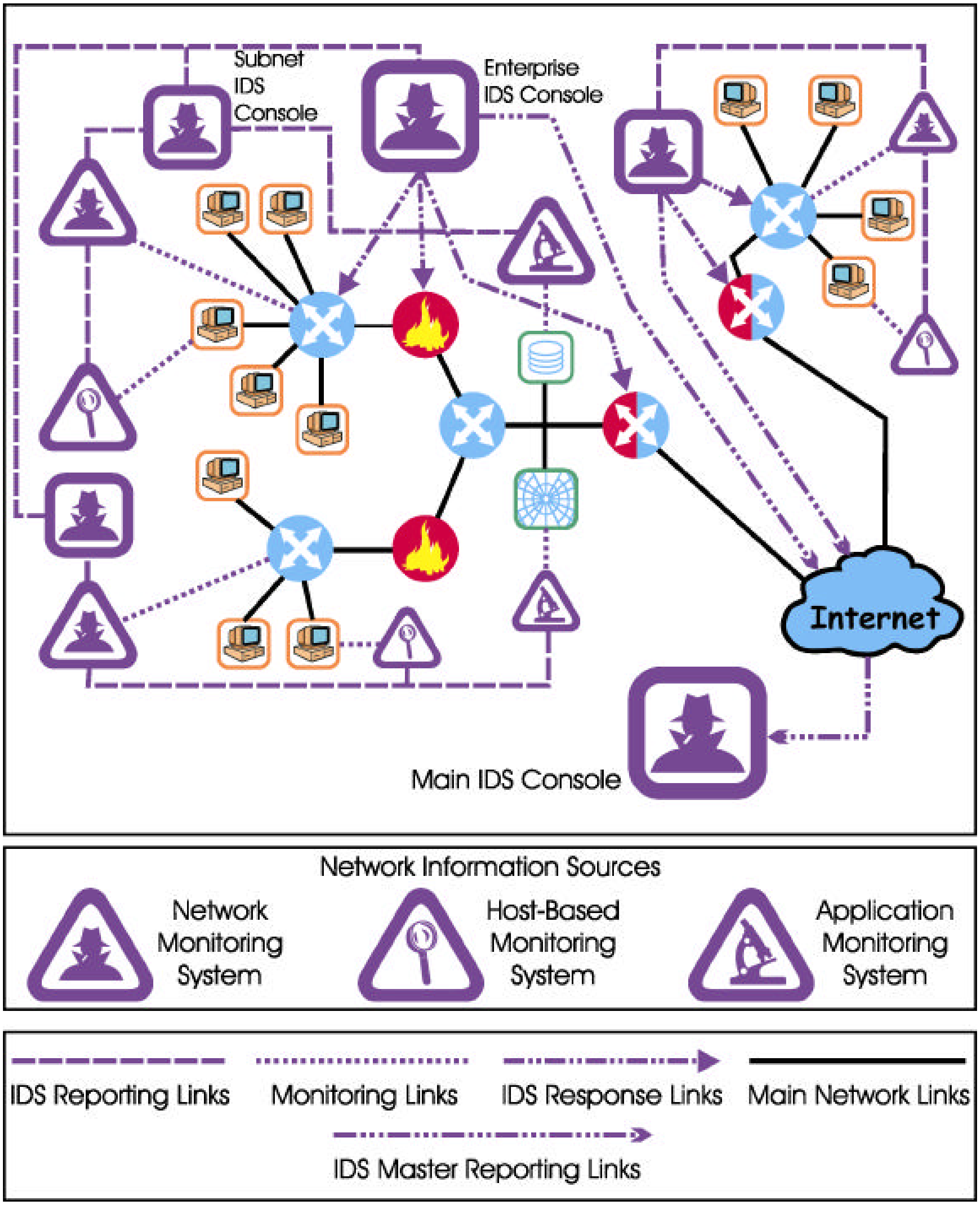
Subnet IDS Console
Enterprise IDS Console
Internet
Main IDS Console
Network Information Sources
Network Monitoring System
Host-Based Monitoring System
Application Monitoring System
IDS Reporting Links
Monitoring Links
IDS Response Links
Main Network Links
IDS Master Reporting Links

Figure 2: Distributed Control Strategy

2.3.5.3. Fully Distributed (Figure 3)

Monitoring and detection is done using an agent-based approach, where response decisions are made at the point of analysis.

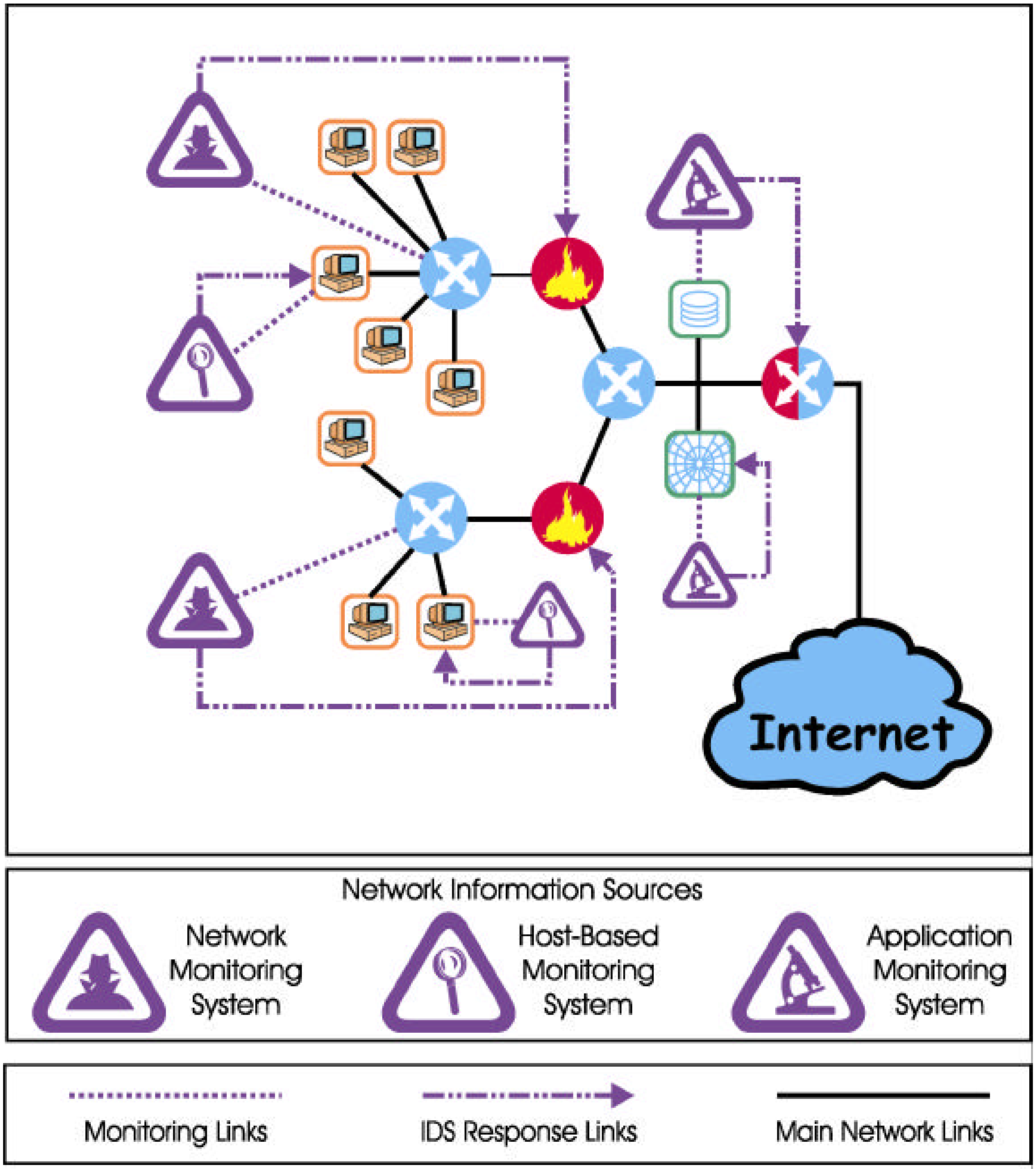
Internet
Network Information Sources
Network Monitoring System
Host-Based Monitoring System
Application Monitoring System
Monitoring Links
IDS Response Links
Main Network Links

Figure 3: Fully Distributed (Agent-Based) Control

2.3.6. Timing

Timing refers to the elapsed time between the events that are monitored and the analysis of those events.

2.3.6.1. Interval-Based (Batch Mode)

In interval-based IDSs, the information flow from monitoring points to analysis engines is not continuous. In effect, the information is handled in a fashion similar to “store and forward” communications schemes. Many early host-based IDSs used this timing scheme, as they relied on operating system audit trails, which were generated as files. Interval-based IDSs are precluded from performing active responses.

2.3.6.2. Real-Time(Continuous)

Real-time IDSs operate on continuous information feeds from information sources. This is the predominant timing scheme for network-based IDSs, which gather information from network traffic streams. In this document, we use the term “real-time” as it is used in process control situations. This means that detection performed by a “real-time” IDS yields results quickly enough to allow the IDS to take action that affects the progress of the detected attack.

2.3.7. Information Sources

The most common way to classify IDSs is to group them by information source. Some IDSs analyze network packets, captured from network backbones or LAN segments, to find attackers. Other IDSs analyze information sources generated by the operating system or application software for signs of intrusion.

2.3.7.1. Network-Based IDSs

The majority of commercial intrusion detection systems are network-based. These IDSs detect attacks by capturing and analyzing network packets. Listening on a network segment or switch, one network-based IDS can monitor the network traffic affecting multiple hosts that are connected to the network segment, thereby protecting those hosts.

Network-based IDSs often consist of a set of single-purpose sensors or hosts placed at various points in a network. These units monitor network traffic, performing local analysis of that traffic and reporting attacks to a central management console. As the sensors are limited to running the IDS, they can be more easily secured against attack. Many of these sensors are designed to run in “stealth” mode, in order to make it more difficult for an attacker to determine their presence and location.

Advantages of Network-Based IDSs:

- A few well-placed network-based IDSs can monitor a large network.
- The deployment of network-based IDSs has little impact upon an existing network. Network-based IDSs are usually passive devices that listen on a network wire without interfering with the normal operation of a network. Thus, it is usually easy to retrofit a network to include network-based IDSs with minimal effort.
- Network-based IDSs can be made very secure against attack and even made invisible to many attackers.

Disadvantages of Network-Based IDSs:

- Network-based IDSs may have difficulty processing all packets in a large or busy network and, therefore, may fail to recognize an attack launched during periods of high traffic. Some vendors are attempting to solve this problem by implementing IDSs completely in hardware, which is much faster. The need to analyze packets quickly also forces vendors to both detect fewer attacks and also detect attacks with as little computing resource as possible which can reduce detection effectiveness.
- Many of the advantages of network-based IDSs don't apply to more modern switch-based networks. Switches subdivide networks into many small segments (usually one fast Ethernet wire per host) and provide dedicated links between hosts serviced by the same switch. Most switches do not provide universal monitoring ports and this limits the monitoring range of a network-based IDS sensor to a single host. Even when switches provide such monitoring ports, often the single port cannot mirror all traffic traversing the switch.
- Network-based IDSs cannot analyze encrypted information. This problem is increasing as more organizations (and attackers) use virtual private networks.
- Most network-based IDSs cannot tell whether or not an attack was successful; they can only discern that an attack was initiated. This means that after a network-based IDS detects an attack, administrators must manually investigate each attacked host to determine whether it was indeed penetrated.
- Some network-based IDSs have problems dealing with network-based attacks that involve fragmenting packets. These malformed packets cause the IDSs to become unstable and crash.

2.3.7.2. Host-Based IDSs

Host-based IDSs operate on information collected from within an individual computer system. (Note that application-based IDSs are actually a subset of host-based IDSs.) This vantage point allows host-based IDSs to analyze activities with great reliability and precision, determining exactly which processes and users are involved in a particular attack on the operating system. Furthermore, unlike network-based IDSs, host-based IDSs can "see" the outcome of an attempted attack, as they can directly access and monitor the data files and system processes usually targeted by attacks.

Host-based IDSs normally utilize information sources of two types, operating system audit trails, and system logs. Operating system audit trails are usually generated at the innermost (kernel) level of the operating system, and are therefore more detailed and better protected than system logs. However, system logs are much less obtuse and much smaller than audit trails, and are furthermore far easier to comprehend. Some host-based IDSs are designed to support a centralized IDS management and reporting infrastructure that can allow a single management console to track many hosts. Others generate messages in formats that are compatible with network management systems.

Advantages:

- Host-based IDSs, with their ability to monitor events local to a host, can detect attacks that cannot be seen by a network-based IDS.
- Host-based IDSs can often operate in an environment in which network traffic is encrypted, when the host-based information sources are generated before data is encrypted and/or after the data is decrypted at the destination host
- Host-based IDSs are unaffected by switched networks.
- When Host-based IDSs operate on OS audit trails, they can help detect Trojan Horse or other attacks that involve software integrity breaches. These appear as inconsistencies in process execution.

Disadvantages:

- Host-based IDSs are harder to manage, as information must be configured and managed for every host monitored.
- Since at least the information sources (and sometimes part of the analysis engines) for host-based IDSs reside on the host targeted by attacks, the IDS may be attacked and disabled as part of the attack.
- Host-based IDSs are not well suited for detecting network scans or other such surveillance that targets an entire network, because the IDS only sees those network packets received by its host.
- Host-based IDSs can be disabled by certain denial-of-service attacks.
- When host-based IDSs use operating system audit trails as an information source, the amount of information can be immense, requiring additional local storage on the system.
- Host-based IDSs use the computing resources of the hosts they are monitoring, therefore inflicting a performance cost on the monitored systems.

2.3.7.3. Application-Based IDSs

Application-based IDSs are a special subset of host-based IDSs that analyze the events transpiring within a software application. The most common information sources used by application-based IDSs are the application’s transaction log files.

The ability to interface with the application directly, with significant domain or application-specific knowledge included in the analysis engine, allows application-based IDSs to detect suspicious behavior due to authorized users exceeding their authorization. This is because such problems are more likely to appear in the interaction between the user, the data, and the application.

Advantages:

- Application-based IDSs can monitor the interaction between user and application, which often allows them to trace unauthorized activity to individual users.
- Application-based IDSs can often work in encrypted environments, since they interface with the application at transaction endpoints, where information is presented to users in unencrypted form.

Disadvantages:

- Application-based IDSs may be more vulnerable than host-based IDSs to attacks as the applications logs are not as well-protected as the operating system audit trails used for host-based IDSs.
- As Application-based IDSs often monitor events at the user level of abstraction, they usually cannot detect Trojan Horse or other such software tampering attacks. Therefore, it is advisable to use an Application-based IDS in combination with Host-based and/or Network-based IDSs.

2.3.8. IDS Analysis

There are two primary approaches to analyzing events to detect attacks: misuse detection and anomaly detection. Misuse detection, in which the analysis targets something known to be “bad”, is the technique used by most commercial systems. Anomaly detection, in which the analysis looks for abnormal patterns of activity, has been, and continues to be, the subject of a great deal of research. Anomaly detection is used in limited form by a number of IDSs. There are strengths and weaknesses associated with each approach, and it appears that the most effective IDSs use mostly misuse detection methods with a smattering of anomaly detection components.

2.3.8.1. Misuse Detection

Misuse detectors analyze system activity, looking for events or sets of events that match a predefined pattern of events that describe a known attack. As the patterns corresponding to known attacks are called *signatures*, misuse detection is sometimes called “signature-based detection.” The most common form of misuse detection used in commercial products specifies each pattern of events corresponding to an attack as a separate signature. However, there are more sophisticated approaches to doing misuse detection (called “state-based” analysis techniques) that can leverage a single signature to detect groups of attacks.

Advantages:

- Misuse detectors are very effective at detecting attacks without generating an overwhelming number of false alarms.

- Misuse detectors can quickly and reliably diagnose the use of a specific attack tool or technique. This can help security managers prioritize corrective measures.
- Misuse detectors can allow system managers, regardless of their level of security expertise, to track security problems on their systems, initiating incident handling procedures.

Disadvantages:

- Misuse detectors can only detect those attacks they know about – therefore they must be constantly updated with signatures of new attacks.
- Many misuse detectors are designed to use tightly defined signatures that prevent them from detecting variants of common attacks. State-based misuse detectors can overcome this limitation, but are not commonly used in commercial IDSs.

2.3.8.2. Anomaly Detection

Anomaly detectors identify abnormal unusual behavior (anomalies) on a host or network. They function on the assumption that attacks are different from “normal” (legitimate) activity and can therefore be detected by systems that identify these differences. Anomaly detectors construct profiles representing normal behavior of users, hosts, or network connections. These profiles are constructed from historical data collected over a period of normal operation. The detectors then collect event data and use a variety of measures to determine when monitored activity deviates from the norm.

The measures and techniques used in anomaly detection include:

- Threshold detection, in which certain attributes of user and system behavior are expressed in terms of counts, with some level established as permissible. Such behavior attributes can include the number of files accessed by a user in a given period of time, the number of failed attempts to login to the system, the amount of CPU utilized by a process, etc. This level can be static or heuristic (*i.e.*, designed to change with actual values observed over time)
- Statistical measures, both parametric, where the distribution of the profiled attributes is assumed to fit a particular pattern, and non-parametric, where the distribution of the profiled attributes is “learned” from a set of historical values, observed over time.
- Rule-based measures, which are similar to non-parametric statistical measures in that observed data defines acceptable usage patterns, but differs in that those patterns are specified as rules, not numeric quantities
- Other measures, including neural networks, genetic algorithms, and immune system models.

Only the first two measures are used in current commercial IDSs.

Unfortunately, anomaly detectors and the IDSs based on them often produce a large number of false alarms, as normal patterns of user and system behavior can vary wildly. Despite this shortcoming, researchers assert that anomaly-based IDSs are able to detect new attack forms, unlike signature-based IDSs that rely on matching patterns of past attacks.

Furthermore, some forms of anomaly detection produce output that can in turn be used as information sources for misuse detectors. For example, a threshold-based anomaly detector can generate a figure representing the "normal" number of files accessed by a particular user; the misuse detector can use this figure as part of a detection signature that says "if the number of files accessed by this user exceeds this "normal" figure by ten percent, trigger an alarm."

Although some commercial IDSs include limited forms of anomaly detection, few, if any, rely solely on this technology.The anomaly detection that exists in commercial systems usually revolves around detecting network or port scanning. However, anomaly detection remains an active intrusion detection research area and may play a greater part in future IDSs.

Advantages:

- IDSs based on anomaly detection detect unusual behavior and thus have the ability to detect symptoms of attacks without specific knowledge of details.
- Anomaly detectors can produce information that can in turn be used to define signatures for misuse detectors.

Disadvantages:

- Anomaly detection approaches usually produce a large number of false alarms due to the unpredictable behaviors of users and networks.
- Anomaly detection approaches often require extensive "training sets" of system event records in order to characterize normal behavior patterns.

2.3.9. Response Options for IDSs

Once IDSs have obtained event information and analyzed it to find symptoms of attacks, they generate responses. Some of these responses involve reporting results and findings to a pre-specified location. Others involve more active automated responses. Though researchers are tempted to underrate the importance of good response functions in IDSs, they are actually very important. Commercial IDSs support a wide range of response options, often categorized as active responses, passive responses, or some mixture of the two.

2.3.9.1. Active Responses

Active IDS responses are automated actions taken when certain types of intrusions are detected. There are three categories of active responses.

Collect additional information

The most innocuous, but at times most productive, active response is to collect additional information about a suspected attack. Each of us have probably done the equivalent of this when awakened by a strange noise at night. The first thing one does in such a situation is to listen more closely, searching for additional information that allows you to decide whether you should take action.

In the IDS case, this might involve increasing the level of sensitivity of information sources (for instance, turning up the number of events logged by an operating system audit trail, or increasing the sensitivity of a network monitor to capture all packets, not just those targeting a particular port or target system.) Collecting additional information is helpful for several reasons. The additional information collected can help resolve the detection of the attack (assisting the system in diagnosing whether an attack did or did not take place). This option also allows the organization to gather information that can be used to support investigation and apprehension of the attacker, and to support criminal and civil legal remedies.

Change the Environment

Another active response is to halt an attack in progress and then block subsequent access by the attacker. Typically, IDSs do not have the ability to block a specific person's access, but instead block Internet Protocol (IP) addresses from which the attacker appears to be coming. It is very difficult to block a determined and knowledgeable attacker, but IDSs can often deter expert attackers or stop novice attackers by taking the following actions:

- Injecting TCP reset packets into the attacker's connection to the victim system, thereby terminating the connection
- Reconfiguring routers and firewalls to block packets from the attacker's apparent location (IP address or site),
- Reconfiguring routers and firewalls to block the network ports, protocols, or services being used by an attacker, and
- In extreme situations, reconfiguring routers and firewalls to sever all connections that use certain network interfaces.

Take Action Against the Intruder

Some who follow intrusion detection discussions, especially in information warfare circles, believe that the first option in active response is to take action against the intruder. The most aggressive form of this response involves launching attacks against or attempting to actively gain information about the attacker's host or site. However tempting it might be, this response is ill advised. Due to legal ambiguities about civil liability, this option can represent a greater risk than the attack it is intended to block.

The first reason for approaching this option with a great deal of caution is that it may be illegal. Furthermore, as many attackers use false network addresses when attacking systems, it carries with it a high risk of causing damage to innocent Internet sites and users. Finally, strike back can escalate the attack, provoking an attacker who originally intended only to browse a site to take more aggressive action.

Should an active intervention and traceback of this sort be warranted (as in the case of a critical system) human control and supervision of the process is advisable. We strongly recommend that you obtain legal advice before pursuing any of these “strike-back” options.

2.3.9.2. Passive Responses

Passive IDS responses provide information to system users, relying on humans to take subsequent action based on that information. Many commercial IDSs rely solely on passive responses.

Alarms and Notifications

Alarms and notifications are generated by IDSs to inform users when attacks are detected. Most commercial IDSs allow users a great deal of latitude in determining how and when alarms are generated and to whom they are displayed.

The most common form of alarm is an onscreen alert or popup window. This is displayed on the IDS console or on other systems as specified by the user during the configuration of the IDS. The information provided in the alarm message varies widely, ranging from a notification that an intrusion has taken place to extremely detailed messages outlining the IP addresses of the source and target of the attack, the specific attack tool used to gain access, and the outcome of the attack.

Another set of options that are of utility to large or distributed organizations are those involving remote notification of alarms or alerts. These allow organizations to configure the IDS so that it sends alerts to cellular phones and pagers carried by incident response teams or system security personnel.

Some products also offer email as another notification channel. This is ill advised, as attackers often routinely monitor email and might even block the message.

SNMP Traps and Plug-ins

Some commercial IDSs are designed to generate alarms and alerts, reporting them to a network management system. These use SNMP traps and messages to post alarms and alerts to central network management consoles, where they can be serviced by network operations personnel. Several benefits are associated with this reporting scheme, including the ability to adapt the entire network infrastructure to respond to a detected attack, the ability to shift the processing load associated with an active response to a system other than the one being targeted by the attack, and the ability to use common communications channels.

2.3.9.3. Reporting and Archiving Capabilities

Many, if not all, commercial IDSs provide capabilities to generate routine reports and other detailed information documents. Some of these can output reports of system events and intrusions detected over a particular reporting period (for example, a week or a month.) Some provide statistics or logs generated by the IDS in formats suitable for inclusion in database systems or for use in report generating packages (An example of such a commonly-supported package is Crystal Reports.)

2.3.9.4. Failsafe considerations for IDS responses

When identifying candidate IDSs for your organization, it is important to consider the failsafe features included by the IDS vendor. Failsafe features are those design features meant to protect the IDS from being circumvented or defeated by an attacker. These represent a necessary difference between standard system management tools and security management tools.

There are several areas in the response function that require failsafe measures. For instance, IDSs need to provide silent, reliable monitoring of attackers. Should the response function of an IDS break this silence by broadcasting alarms and alerts in plaintext over the monitored network, it would allow attackers to detect the presence of the IDS. Worse yet, the attackers can directly target the IDS as part of the attack on the victim system.

Encrypted tunnels or other cryptographic measures used to hide and authenticate IDS communications are excellent ways to secure and ensure the reliability of the IDS.

2.4. Tools that Complement IDSs

Several tools exist that complement IDSs and are often labeled as intrusion detection products by vendors since they perform similar functions. This section discusses four of these tools, Vulnerability Analysis Systems, File Integrity Checkers, Honey Pots, and Padded Cells, and describes how they can enhance an organization’s intrusion detection capability.

2.4.1. Vulnerability Analysis or Assessment Systems

Vulnerability analysis (also known as vulnerability assessment) tools test to determine whether a network or host is vulnerable to known attacks. Vulnerability assessment represents a special case of the intrusion detection process. The information sources used are system state attributes and outcomes of attempted attacks. The information sources are collected by a part of the assessment engine. The timing of analysis is interval-based or batch-mode, and the type of analysis is misuse detection. This means that vulnerability assessment systems are essentially batch mode misuse detectors that operate on system state information and results of specified test routines.

Vulnerability analysis is a very powerful security management technique, but is suitable as a complement to using an IDS, not as a replacement. Should an organization rely solely on vulnerability analysis tools to monitor systems, a knowledgeable attacker may monitor the vulnerability analysis system, note when the information source is collected, and time the attack to fit between collection times.

However, vulnerability analysis systems can reliably generate a “snapshot” of the security state of a system at a particular time. Furthermore, as they exhaustively test systems for vulnerability to large numbers of known attacks, vulnerability analysis systems can allow a security manager to check for problems due to human error or to audit the system for compliance with a particular system security policy.

2.4.1.1. Vulnerability Analysis System Process

The general process for vulnerability assessment is as follows:

- A specified set of system attributes is sampled
- The results of the sampling are stored in a secure data repository
- The results are organized and compared to at least one reference set of data (this set can be a manually specified “ideal configuration” template or a snapshot of the system state generated earlier)
- Any differences between the two sets are identified and reported.

Commercial vulnerability assessment products often optimize this process by:

- splitting processing loads, running multiple assessment engines in parallel.
- using cryptographic mechanisms to do very sensitive and reliable tests of whether particular files or objects have changed unexpectedly.

2.4.1.2. Vulnerability Analysis Types

There are two major ways of classifying vulnerability analysis systems, first, by the location from which assessment information is gathered, and second, by the assumptions regarding the level of trust invested in the assessment tool. Those who use the first classification scheme for vulnerability assessment classify systems as either *network-based* or *host-based*. Those who use the second classification scheme, classify systems as *credentialed* or *non-credentialed*. These terms refer to whether the analysis is done with or without system credentials (such as passwords or other identification and authentication that grant access to the system internals.) In this paper, we will use the first classification scheme to describe the different approaches for vulnerability analysis.

Host-based Vulnerability Analysis

Host-based vulnerability analysis systems determine vulnerability by assessing system data sources such as file contents, configuration settings, and other status information. This information is usually accessible using standard system queries and inspection of system attributes. As the information is gathered under the assumption that the vulnerability analyzer is granted access to the host, it is also sometimes known as *credential-based* vulnerability assessment. This class of assessment is also labeled *passive* assessment.

The vulnerabilities best revealed by host-based vulnerability assessment are those involving privilege escalation attacks. (Such attacks might seek *superuser* or *root* privilege on a UNIX system, or *administrator* access on an NT system.)

Network-Based Vulnerability Analysis

Network-based vulnerability analysis systems have gained acceptance in recent years. These vulnerability analysis systems require a remote connection to the target system. They may actually reenact system attacks, noting and recording responses to these attacks or simply probe different targets to infer weaknesses from their responses. This reenactment of attacks or probing can occur regardless of whether one has permission to access the target system; hence this is considered *non-credentialed* assessment. Furthermore, as network-based vulnerability analysis is defined as actively attacking or scanning the targeted system, it is also sometimes labeled *active* vulnerability assessment.

Network-based vulnerability analysis tools are sometimes marketed as intrusion detection tools. Although, as discussed earlier in this document, this is correct by some definitions of intrusion detection, a vulnerability analysis product is not a complete intrusion detection solution for most environments.

There are two methods typically used in network-based vulnerability assessment:

- *Testing by exploit* – in this method, the system reenacts an actual attack. A status flag is returned indicating whether the attack was successful.
- *Inference Methods* – in this method, the system doesn't actually exploit vulnerabilities, but looks for the artifacts that successful attacks would leave behind. Examples of inference techniques involve checking version numbers provided by systems as results of queries, checking ports to determine which are open, and checking protocol compliance by making simple requests for status or information.

2.4.1.3. Advantages and Disadvantages of Vulnerability Analysis

Advantages

- Vulnerability Analysis is of significant value as a part of a security monitoring system, allowing the detection of problems on systems that cannot support an IDS.
- Vulnerability Analysis Systems provide security-specific testing capabilities for documenting the security state of systems at the

start of a security program and for reestablishing the security baseline whenever major changes occur.

- When Vulnerability Analysis Systems are used on a regular schedule, they can reliably spot changes in the security state of a system, alerting security managers to problems that require correction.
- Vulnerability Analysis Systems offer a way for security managers and system administrators to double-check any changes they make to systems, assuring that in mitigating one set of security problems, they do not create another set of problems.

Disadvantages and Issues

- Host-based vulnerability analyzers are tightly bound to specific operating systems and applications; they are therefore often more costly to build, maintain, and manage.
- Network-based vulnerability analyzers are platform-independent, but less accurate and subject to more false alarms.
- Some network-based checks, especially those for denial-of-service attacks, can crash the systems they're testing.
- When conducting vulnerability assessment of networks on which intrusion detection systems are running, the IDSs can block subsequent assessments. Worse yet, repeated network-based assessments can "train" certain anomaly-detection-based IDSs to ignore real attacks.
- Organizations that use vulnerability assessment systems must take care to assure that their testing is limited to systems within their political or management control boundaries. Privacy issues must be taken into account, especially when employee or customer personal data is included in information sources.

2.4.2. File Integrity Checkers

File Integrity Checkers are another class of security tools that complement IDSs. They utilize message digest or other cryptographic checksums for critical files and objects, comparing them to reference values, and flagging differences or changes.

The use of cryptographic checksums is important, as attackers often alter system files, at three stages of the attack. First, they alter system files as the goal of the attack (e.g., Trojan Horse placement), second, they attempt to leave back doors in the system through which they can reenter the system at a later time, and finally, they attempt to cover their tracks so that system owners will be unaware of the attack.

Although File Integrity Checkers are most often used to determine whether attackers have altered system files or executables, they can also help determine whether vendor-supplied bug patches or other desired changes have been applied to system binaries. They are extremely valuable to those conducting a forensic examination of systems that have been attacked, as they allow quick and reliable diagnosis of the footprint of an attack. This enables system managers to optimize the restoration of service after incidents occur.

The freeware product, Tripwire (www.tripwiresecurity.com) is perhaps the best-known example of File Integrity Checkers.

2.4.3. Honey Pot and Padded Cell Systems

Several novel additions to the intrusion detection product line are under development and may soon become available. It is important to understand how these products differ from traditional IDSs and to realize that they are not yet widely used.

Honey pots are decoy systems that are designed to lure a potential attacker away from critical systems. Honey pots are designed to:

- divert an attacker from accessing critical systems,
- collect information about the attacker's activity, and
- encourage the attacker to stay on the system long enough for administrators to respond.

These systems are filled with fabricated information designed to appear valuable but that a legitimate user of the system wouldn't access. Thus, any access to the honey pot is suspect. The system is instrumented with sensitive monitors and event loggers that detect these accesses and collect information about the attacker's activities.

Padded cells take a different approach. Instead of trying to attract attackers with tempting data, a padded cell operates in tandem with a traditional IDS. When the IDS detects attackers, it seamlessly transfers then to a special padded cell host. Once the attackers are in the padded cell, they are contained within a simulated environment where they can cause no harm. As in honey pots, this simulated environment can be filled with interesting data designed to convince an attacker that the attack is going according to plan. As in honey pots, padded cells are well-instrumented and offer unique opportunities to monitor the actions of an attacker. IDS researchers have used padded cell and honey pot systems since the late 1980s, but until recently no commercial products have been under development. It is important to seek guidance from legal counsel before deciding to use either of these systems in your operational environment.

Advantages:

- Attackers can be diverted to system targets that they cannot damage.
- Administrators have additional time to decide how to respond to an attacker.
- Attackers' actions can be easily and more extensively monitored, with results used to refine threat models and improve system protections.
- Honey pots may be effective at catching insiders who are snooping around a network.

Disadvantages:

- The legal implications of using such devices are not well defined
- Honey pots and padded cells have not yet been shown to be generally useful security technologies.
- An expert attacker, once diverted into a decoy system, may become angry and launch a more hostile attack against an organization's systems.
- A high level of expertise is needed for administrators and security managers in order to use these systems.

3. Advice on selecting IDS products

The wide array of intrusion detection products available today addresses a range of organizational security goals and considerations. Given this range of products and features, the process of selecting products that represent the best fit for your organization's needs is, at times, difficult. The following questions may be used as guidance when preparing a specification for acquiring an intrusion detection product.

3.1. Technical and Policy Considerations

In order to determine which IDSs can be used in your environment, you must first consider that environment, in technical, physical, and political terms.

3.1.1. What is your system environment?

The first hurdle an IDS must clear is that of functioning in your systems environment. This is important, for if an IDS is not designed to accommodate the information sources that are available on your systems, it will not be able to see anything that goes on in your systems, attack or normal activity.

3.1.1.1. What are the technical specifications of your systems environment?

First, specify the technical attributes of your systems environment. Examples of information specified here would include network diagrams and maps specifying the number and locations of hosts, operating systems for each host, the number and types of network devices such as routers, bridges, and switches, number and types of terminal servers and dialup connections, and descriptors of any network servers, including types, configurations, and application software and versions running on each. If you run any enterprise network management system, specify it here.

3.1.1.2. What are the technical specifications of your current security protections?

Once you have described the technical attributes of your systems environment, describe the security protections you already have in place.

Specify numbers, types, and locations of network firewalls, identification and authentication servers, data and link encryptors, anti-virus packages, access control products, specialized security hardware (such as crypto accelerator hardware for web servers), virtual private networks, and any other security mechanisms on your systems.

3.1.1.3. What are the goals of your enterprise?

Some IDSs have been developed to accommodate the special needs of certain industries or market niches such as electronic commerce, health care, or financial markets. Define the functional goals of your enterprise (there can be several goals associated with a single organization) that are supported by your systems.

3.1.1.4. How formal is the system environment and management culture in your organization?

Organizational styles vary, depending on the function of the organization and its traditional culture. For instance, military or other organizations that deal with national security issues tend to operate with a high degree of formality, especially when contrasted with university or other academic environments.

Some IDSs offer features that support enforcement of formal use policies, with configuration screens that accept formal expressions of policy, and extensive reporting capabilities that do detailed reporting of policy violations.

3.1.2. What are your security goals and objectives?

Once you’ve specified the technical landscape of your organizations systems as well as the existing security mechanisms, it’s time to articulate the goals and objectives you wish to attain by using an IDS.

3.1.2.1. Is the primary concern of your organization protecting from threat originating outside your organization?

Perhaps the easiest way to specify security goals is by categorizing your organization’s threat concerns. First, state, as specifically as possible, the concerns that your organization has regarding threat that originates outside the organization.

3.1.2.2. Is your organization concerned about insider attack?

Repeat the last step, this time addressing concerns about threat that originates from within your organization, encompassing not only the user who attacks the system from within (such as a shipping clerk who attempts to access and alter the payroll system) but also the authorized user who overstep their privileges thereby violating organizational security policy or laws (customer service agents who, driven by curiosity, access earnings and payroll records for public figures.)

3.1.2.3. Does your organization want to use the output of your IDS to determine new needs?

System usage monitoring is sometimes provided as a generic system management tool to determine when system assets require upgrading or replacement. When such monitoring is performed by an IDS, the needs for upgrade can show up as anomalous levels of user activity.

3.1.2.4. Does your organization want to use the IDS to maintain managerial control (non-security related) over network usage?

In some organizations, there are system use policies that target user behaviors that may be classified as personnel management rather than system security issues. These might include accessing web sites that provide content of questionable taste or value (such as pornography) or using organizational systems to send email or other messages for the purpose of harassing individuals. Some IDSs provide features that accommodate detecting such violations of management controls.

3.1.3. What is your existing security policy?

At this time, you should review your existing organization security policy. This will serve as the template against which features of your IDS will be configured. As such, you may find you need to augment the policy, or else derive the following items from it.

3.1.3.1. How is it structured?

It is helpful to articulate the goals outlined in the security policy in terms of the standard security goals (integrity, confidentiality, and availability) as well as more generic management goals (privacy, protection from liability, manageability.)

3.1.3.2. What are the general job descriptions of your system users?

List the general job functions of system users (there are commonly several functions assigned to a single user) as well as the data and network accesses that each function requires.

3.1.3.3. Does the policy include reasonable use policies or other management provisions?

As mentioned above, many organizations have system use policies included as part of security policies.

3.1.3.4. Has your organization defined processes for dealing with specific policy violations?

It is helpful to have a clear idea of what the organization wishes to do when the IDS detects that a policy has been violated. If the organization doesn’t intend to react to such violations, it may not make sense to configure the IDS to detect them. If, on the other hand, the organization wishes to actively respond to such violations, the IDS operational staff should be informed of the organization’s response policy so that they can deal with alarms in an appropriate manner.

3.2. Organizational Requirements and Constraints

Your organization's operational goals, constraints, and culture will affect the selection of IDSs and other security tools and technologies to protect your systems. In this section, consider these organizational requirements and limitations.

3.2.1. What are requirements that are levied from outside the organization?

Is your organization subject to oversight or review by another organization? If so, does that oversight authority require IDSs or other specific system security resources?

3.2.1.1. Are there requirements for public access to information on your organization's systems?

Do regulations or statutes require that information on your system be accessible by the public during certain hours of the day, or during certain date or time intervals?

3.2.1.2. Are there other security-specific requirements levied by law?

Are there legal requirements for protection of personal information (such as earnings information or medical records) stored on your systems? Are there legal requirements for investigation of security violations that divulge or endanger that information?

3.2.1.3. Are there internal audit requirements for security best practices or due diligence?

Do any of these audit requirements specify functions that the IDS must provide or support?

3.2.1.4. Is the system subject to accreditation?

If so, what is the accreditation authority's requirement for IDS or other security protection?

3.2.1.5. Are there requirements for law enforcement investigation and resolution of security incidents?

Do these specify any IDS functions, especially those having to do with collection and protection of IDS logs as evidence?

3.2.2. What are your organization's resource constraints?

IDSs can protect the systems of an organization, but at a price. It makes little sense to incur additional expense for IDS features if your organization does not have sufficient systems or personnel to use them.

3.2.2.1. What is the budget for acquisition and life cycle support of intrusion detection hardware, software, and infrastructure?

Remember here that the acquisition of IDS software is not the total cost of ownership; you may also have to acquire a system on which to run the software, specialized assistance in installing and configuring the system, and training your personnel.

3.2.2.2. Is there sufficient existing staff to monitor an intrusion detection system full time?

Some IDSs are designed under the assumption that systems personnel will attend them around the clock. If you do not anticipate having such personnel available, you may wish to explore those systems that accommodate less than full-time attendance or else consider systems that are designed for unattended use.

3.2.2.3. Does your organization have authority to instigate changes based on the findings of an intrusion detection system?

It is critical that you and your organization be clear about what you plan to do with the problems uncovered by an IDS. If you are not empowered to handle the incidents that arise as a result of the monitoring, you should consider coordinating your selection and configuration of the IDS with the party who is.

3.3. IDS Product Features and Quality

3.3.1. Is the product sufficiently scalable for your environment?

As mentioned before in this document, many IDSs are not able to scale to large or widely distributed enterprise network environments.

3.3.2. How has the product been tested?

Simply asserting that an IDS has certain capabilities is not sufficient to demonstrate that those capabilities are real. You should request additional demonstration of the suitability of a particular IDS to your environment and goals.

3.3.2.1. Has the product been tested against functional requirements?

Ask the vendor about the assumptions made regarding the goals and constraints of customer environments.

3.3.2.2. Has the product been tested against attack?

Ask vendors for details of the security testing to which its products have been subjected. If the product includes network-based vulnerability assessment features, ask also whether test routines that produce system crashes or other denials of service have been identified and flagged in system documentation and interfaces.

3.3.3. What is the user level of expertise targeted by the product?

Different IDS vendors target users with different levels of technical and security expertise. Ask the vendor what their assumptions are regarding the users of their products.

3.3.4. Is the product designed to evolve as the organization grows?

One product design goal that will enhance its value to your organization over time is the ability to adapt to your needs over time.

3.3.4.1. Can the product adapt to growth in user expertise?

Ask here whether the IDS interface can be configured (with shortcut keys, customizable alarm features, and custom signatures) on the fly. Ask also whether these features are documented and supported.

3.3.4.2. Can the product adapt to growth and change of the organization's systems infrastructure?

This question has to do with the ability of the IDS to scale to an expanding and increasingly diverse network. Most vendors have experience in adapting their products as target networks grow. Ask also about commitments to support new protocol standards and platform types.

3.3.4.3. Can the product adapt to growth and change of the security threat environment?

This question is especially critical given the current Internet threat environment, in which 30-40 new attacks are posted to the Web every month.

3.3.5. What are the support provisions for the product?

Like other systems, IDSs require maintenance and support over time. In this section, these needs are specified.

3.3.5.1. What are commitments for product installation and configuration support?

Many vendors provide expert assistance to customers in installing and configuring IDSs; others expect that your own staff will handle these functions, and provide only telephone or email help desk functions.

3.3.5.2. What are commitments for ongoing product support?

In this area, ask about the vendor's commitment to supporting your use of their IDS product.

Are subscriptions to signature updates included?

As most IDSs are misuse-detectors, the value of the product is only as good as the signature database against which events are analyzed. Most vendors provide subscriptions to signature updates for some period of time (a year is typical.)

How often are subscriptions updated?

In today's threat environment, in which 30-40 new attacks are published every month, this is a critical question.

How quickly after a new attack is made public will the vendor ship a new signature?

If you are using IDSs to protect highly visible or heavily traveled Internet sites, it is especially critical that you receive the signatures for new attacks as soon as possible.

Are software updates included?

Most IDSs are software products and therefore subject to bugs and revisions. Ask the vendor about software update and bug patch support, and determine to what extent they are included in the product you purchase.

How quickly will software updates and patches be issued after a problem is reported to the vendor?

As software bugs in IDSs can allow attackers to nullify their protective effect, it is extremely important that problems be fixed, reliably and quickly.

Are technical support services included? What is the cost?

In this category, technical support services mean vendor assistance in tuning or adapting your IDS to accommodate special needs, be they monitoring a custom or legacy system within your enterprise, or reporting IDS results in a custom protocol or format.

What are the contact provisions for contacting technical support (email, telephone, online chat, web-based reporting)?

The contact provisions will likely tell you whether these technical support services are accessible enough to support incident handling or other time-sensitive needs.

Are there any guarantees associated with the IDS?

As in other software products, IDSs have traditionally had few guarantees associated with them; however, in an attempt to gain market share, some vendors are initiating guarantee programs.

3.3.5.3. What training resources does the vendor provide as part of the product?

Once an IDS is selected, installed, and configured, it must still be operated by your personnel. In order for these people to make optimal use of the IDS, they should be trained in its use. Some vendors provide this training as part of the product package.

3.3.5.4. What additional training resources are available from the vendor and at what cost?

In the case that the IDS vendor does not provide training as part of the IDS package, you should budget appropriately to train your operational personnel.

4. Deploying IDSs

Intrusion detection technology is a necessary addition to every large organization's computer network security infrastructure. However, given the deficiencies of today's intrusion detection products, and the limited security skill level of many system administrators, an effective IDS deployment requires careful planning, preparation, prototyping, testing, and specialized training.

NIST suggests performing a thorough requirements analysis, carefully selecting the intrusion detection strategy and solution that is compatible with the organization's network infrastructure, policies, and resource level.

4.1. Deployment strategy for IDSs

Organizations should consider a staged deployment of IDSs to allow personnel to gain experience and to ascertain how many monitoring and maintenance resources they will require. The resource requirements for each type of IDS vary widely, depending on the organization and systems environment. IDSs require significant preparation and ongoing human interaction. Organizations must have appropriate security policies, plans, and procedures in place so that personnel know how to handle the many and varied alarms IDSs produce.

We recommend consideration of a combination of network-based IDSs and host-based IDSs to protect an enterprise-wide network. We furthermore recommend a staged deployment, starting with network-based IDSs as they are usually the simplest to install and maintain. Next, protect critical servers with host-based IDSs. Utilize vulnerability analysis products on a regular schedule to test IDSs and other security mechanisms for proper function and configuration.

Honey pots and related technologies should be used conservatively and only by organizations with a highly skilled technical staff that are willing to experiment with leading-edge technology. Furthermore, such techniques should be used only after seeking guidance from legal counsel.

4.2. Deploying Network-Based IDSs

One question that arises when deploying network-based IDSs is where to locate the system sensors. There are many options for placing a network-based IDS with different advantages associated with each location:

4.2.1. Location: Behind each external firewall, in the network DMZ

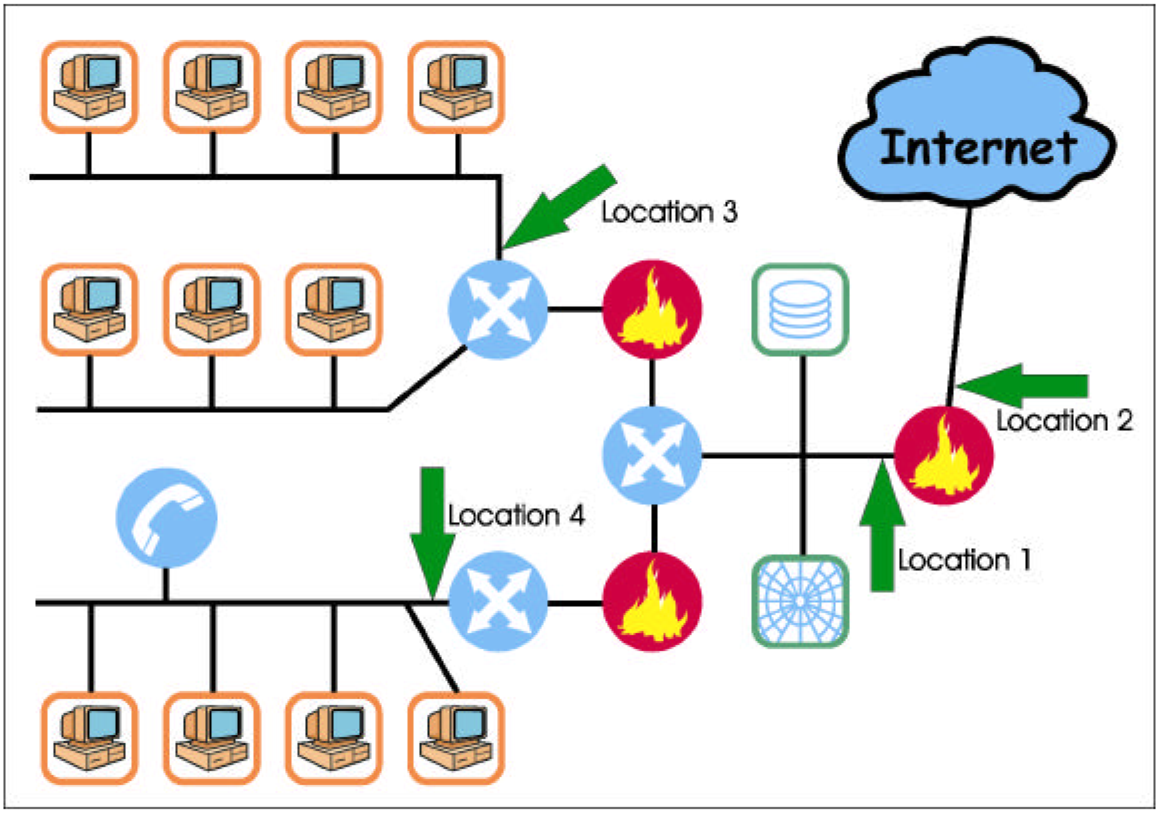

Figure 4 – Locations of Network-based IDS sensors

(See Figure 4 – Location 1)

Advantages:

- Sees attacks, originating from the outside world, that penetrate the network's perimeter defenses.
- Highlights problems with the network firewall policy or performance
- Sees attacks that might target the web server or ftp server, which commonly reside in this DMZ
- Even if the incoming attack is not recognized, the IDS can sometimes recognize the outgoing traffic that results from the compromised server

4.2.2. Location: Outside an external firewall

(See Figure 4 – Location 2)

Advantages:

- Documents number of attacks originating on the Internet that target the network.
- Documents types of attacks originating on the Internet that target the network

4.2.3. Location: On major network backbones

(See Figure 4 – Location 3)

Advantages:

- Monitors a large amount of a network's traffic, thus increasing the possibility of spotting attacks.
- Detects unauthorized activity by authorized users within the organization's security perimeter.

4.2.4. Location: On critical subnets

(See Figure 4 – Location 4)

Advantages:

- Detects attacks targeting critical systems and resources.
- Allows focusing of limited resources to the network assets considered of greatest value.

4.3. Deploying Host-Based IDSs

Once network-based IDSs are in place and operational, the addition of host-based IDSs can offer enhanced levels of protection for your systems. However, installing host-based IDSs on every host in the enterprise can be extremely time-consuming, as each IDS has to be installed and configured for each specific host.

Therefore, we recommend that organizations first install host-based IDSs on critical servers. This will decrease overall deployment costs and allow novice personnel to focus on alarms generated from the most important hosts. Once the operation of host-based IDSs is routine, more security-conscious organizations may consider installing host-based IDSs on the majority of their hosts. In this case, purchase host-based systems that have centralized management and reporting functions. These features will significantly reduce the complexity of managing alerts from a large set of hosts.

Another consideration when using host-based IDSs is that of allowing operators to become familiar with the IDS in a sheltered, but active environment. Much of the effectiveness of any IDS, but particularly a host-based IDS depends on the operator's ability to discern between true and false alarms. Over a period of time, an operator, working with an IDS in a particular environment, will gain a sense of what is normal for that environment, as monitored by the IDS.

It is also important (as host-based IDSs are often not continuously attended by operators) to establish a schedule for checking the results of the IDS. If this is not done, the risk that an adversary will tamper with the IDS in the course of an attack increases.

4.4. Alarm strategies

Finally, when deploying IDSs, the questions of which IDS alarm features to use and when are important issues. Most IDSs come with configurable alarm features, which allow a wide variety of alarm options, including email, paging, network management protocol traps, and even automated blocking of attack sources.

Although these features may be appealing, it is important to be conservative about using them until you have a stable IDS installation and some sense of the behavior of the IDS within your environment. Some experts recommend not activating IDS alarms for as long as several months after installation.

In cases where the alarm and response features include automated response to attacks, specifically those that allow the IDS to direct the firewall to block traffic from the ostensible sources of the attacks, be extremely careful that attackers do not abuse this feature to deny access to legitimate users.

5. Strengths and Limitations of IDSs

Although Intrusion Detection Systems are a valuable addition to an organization’s security infrastructure, there are things they do well, and other things they do not do well. As you plan the security strategy for your organization’s systems, it is important for you to understand what IDSs should be trusted to do and what goals might be better served by other types of security mechanisms.

5.1. Strengths of Intrusion Detection Systems

Intrusion detection systems perform the following functions well:

- Monitoring and analysis of system events and user behaviors
- Testing the security states of system configurations
- Baselining the security state of a system, then tracking any changes to that baseline
- Recognizing patterns of system events that correspond to known attacks
- Recognizing patterns of activity that statistically vary from normal activity
- Managing operating system audit and logging mechanisms and the data they generate
- Alerting appropriate staff by appropriate means when attacks are detected.
- Measuring enforcement of security policies encoded in the analysis engine
- Providing default information security policies
- Allowing non-security experts to perform important security monitoring functions.

5.2. Limitations of Intrusion Detection Systems

Intrusion detection systems cannot perform the following functions:

- Compensating for weak or missing security mechanisms in the protection infrastructure. Such mechanisms include firewalls, identification and authentication, link encryption, access control mechanisms, and virus detection and eradication.

- Instantaneously detecting, reporting, and responding to an attack, when there is a heavy network or processing load.
- Detecting newly published attacks or variants of existing attacks.
- Effectively responding to attacks launched by sophisticated attackers
- Automatically investigating attacks without human intervention.
- Resisting attacks that are intended to defeat or circumvent them
- Compensating for problems with the fidelity of information sources
- Dealing effectively with switched networks.

6. Advice on dealing with IDS output

6.1. Typical IDS Output

Almost all IDSs will output a small summary line about each detected attack. This summary line typically contains the information fields shown below. See the Glossary (Appendix C) for a definition of any unfamiliar terms.

- time/date,
- sensor IP address,
- vendor specific attack name,
- standard attack name (if one exists),
- source and destination IP address,
- source and destination port numbers, and
- network protocol used by attack.

Many IDSs will also provide a generic description of each type of attack. This description is important as it enables the operator to correctly gauge the impact of the attack.

This description usually contains the following information:

- text description of attack,
- attack severity level,
- type of loss experienced as a result of the attack,
- the type of vulnerability the attack exploits,
- list of software types and version numbers that are vulnerable to the attack,
- patch information so that computers can be made invulnerable to the attack, and
- references to public advisories about the attack or the vulnerability it exploits.

6.2. Handling Attacks

Perhaps the best advice anyone can give regarding successfully handling IDS outputs indicating the detection of an attack is “Be Prepared.” Your organization should have Incident Handling Plans and Procedures, which set forth the organization's procedures for handling security incidents, such as viruses, insider abuse of systems, and attacks.

This Incident Handling Plan and Procedure should, at a minimum, assign roles and responsibilities for all parties within the organization, outline the actions that are to be taken when an incident occurs, and establish schedules and content for training everyone about their responsibilities in the incident handling process. Furthermore, you should make provisions to conduct periodic tests (similar to fire drills) of the procedures, in which all organizational parties step through their specific responsibilities and assignments. Take the time to train your IDS operators on the organization's Incident Handling Procedure. If the Procedure predates the addition of the IDS to your security infrastructure, consider taking the time to revisit it, amending it to reflect the role of the IDS. In particular, key the actions prescribed in the procedure to the messages provided by the IDS.

7. Computer Attacks and Vulnerabilities

Many organizations acquire intrusion detection systems (IDSs) because they know that IDSs are a necessary complement to a comprehensive system security architecture. However, given the relative youth of commercial IDSs, most organizations lack experienced IDS operators. Despite vendors claims about ease of usage, such training or experience is absolutely necessary. An IDS is only as effective as the human operating it.

IDSs user interfaces vary greatly in quality. Some produce responses in the form of cryptic text logs while others provide graphical depictions of the attacks on the network. Despite this wide variance in display techniques, most IDSs output the same basic information about computer attacks. If users understand this common set of outputs, they can quickly learn to use the majority of commercial IDSs.

7.1. Attack Types

Most computer attacks only corrupt a system's security in very specific ways. For example, certain attacks may enable an attacker to read specific files but don't allow alteration of any system components. Another attack may allow an attacker to shut down certain system components but doesn't allow access to any files. Despite the varied capabilities of computer attacks, they usually result in violation of only four different security properties: availability, confidentiality, integrity, and control. These violations are described below.

Confidentiality: An attack causes a confidentiality violation if it allows attackers to access data without authorization (either implicit or explicit) from the owner of the information.

Integrity: An attack causes an integrity violation if it allows the (unauthorized) attacker to change the system state or any data residing on or passing through a system

Availability: An attack causes an availability violation if it keeps an authorized user (human or machine) from accessing a particular system resource when, where, and in the form that they need it.

Control: An attack causes a control violation if it grants an (unauthorized) attacker privilege in violation of the access control policy of the system. This privilege enables a subsequent confidentiality, integrity, or availability violation.

7.2. Types of Computer Attacks Commonly Detected by IDSs

Three types of computer attacks are most commonly reported by IDSs: system scanning, denial of service (DOS), and system penetration. These attacks can be launched locally, on the attacked machine, or remotely, using a network to access the target. An IDS operator must understand the differences between these types of attacks, as each requires a different set of responses.

7.2.1. Scanning Attacks

A scanning attack occurs when an attacker probes a target network or system by sending different kinds of packets. (This is similar to the activity described in Section 2.4.1.2 , regarding network-based vulnerability analysis tools. Indeed, the techniques may be identical, but the motive for performing the activity is quite different!)

Using the responses received from the target, the attacker can learn many of the system's characteristics and vulnerabilities. Thus, a scanning attack acts as a target identification tool for an attacker. Scanning attacks do not penetrate or otherwise compromise systems. Various names for the tools used to perform these activities include: network mappers, port mappers, network scanners, port scanners, or vulnerability scanners. Scanning attacks may yield:

- The topology of a target network
- The types of network traffic allowed through a firewall
- The active hosts on the network
- The operating systems those hosts are running
- The server software they are running
- The software version numbers for all detected software

Vulnerability scanners are a special type of scanner that check for specific vulnerabilities in hosts. Thus, an attacker can run a vulnerability scanner and it will output a list of hosts (IP addresses) that are likely to be vulnerable to a specific attack.

With this information, an attacker can precisely identify victim systems on the target network along with specific attacks that can be used to penetrate those systems. Thus, attackers use scanning software to "case" a target before launching a real attack. Unfortunately for victims, just as it is legal for a person to enter a bank and to survey the visible security system, some lawyers say that it is legal for an attacker to scan a host or network. From the perspective of someone performing a scan, they are legally scouring the Internet to find publicly accessible resources.

There are legitimate justifications for scanning activity. Web search engines may scan the Internet looking for new web pages. An individual may scan the Internet looking for free music repositories or for publicly accessible multi-user games. Fundamentally, the same kind of technology that allows one to discover publicly available resources also allows one to analyze a system for security weaknesses (as occurs, as mentioned above, when one uses vulnerability assessment tools). The best IDS signatures for malicious scanning are usually able to discern between legitimate and malicious scanning. Scanning is likely the most common attack as it is the precursor to any serious penetration attempt. If your network is connected to the

Internet, it is almost certain that you are scanned, if not daily, at least a couple of times a week.

7.2.2. Denial of Service Attacks

Denial Of Service (DOS) attacks attempt to slow or shut down targeted network systems or services. In certain Internet communities, DOS attacks are common. For example, Internet Relay Chat users engaged in verbal disputes commonly resort to DOS attacks to win arguments with their opponents. While often used for such trivial purposes, DOS attacks can also be used to shut down major organizations. In well-publicized incidents, DOS attacks were charged with causing major losses to electronic commerce operations, whose customers were unable to access them to make purchases. There are two main types of DOS attacks: flaw exploitation and flooding. It is important for an IDS operator to understand the difference between them.

7.2.2.1. Flaw exploitation DOS Attacks

Flaw exploitation attacks exploit a flaw in the target system’s software in order to cause a processing failure or to cause it to exhaust system resources. An example of such a processing failure is the ‘ping of death’ attack. This attack involved sending an unexpectedly large ping packet to certain Windows systems. The target system could not handle this abnormal packet, and a system crash resulted. With respect to resource exhaustion attacks, the resources targeted include CPU time, memory, disk space, space in a special buffer, or network bandwidth. In many cases, simply patching the software can circumvent this type of DOS attack.

7.2.2.2. Flooding DOS Attacks

Flooding attacks simply send a system or system component more information than it can handle. In cases where the attacker cannot send a system sufficient information to overwhelm its processing capacity, the attacker may nonetheless be able to monopolize the network connection to the target, thereby denying anyone else use of the resource. With these attacks, there is no flaw in the target system that can be patched. This is why such attacks represent a major source of frustration and concern to organizations. While there are few general solutions to stop flooding attacks, there are several technical modifications that can be made by a target to mitigate such an attack.

The term “distributed DOS” (DDOS) is a subset of DOS attacks. DDOS attacks are simply flooding DOS attacks where the attacker uses multiple computers to launch the attack. These attacking computers are centrally controlled by the attacker’s computer and thus act as a single immense attack system. An attacker cannot usually bring down a major e-commerce site by flooding it with network packets from a single host. However, if an attacker gains control of 20,000 hosts and subverts them to run an attack under his direction, then the attacker has a formidable capability to successfully attack the fastest of systems, bringing it to a halt.

7.2.3. Penetration Attacks

Penetration attacks involve the unauthorized acquisition and/or alteration of system privileges, resources, or data. Consider these integrity and control violations as contrasted to DOS attacks that violate the availability of a resource and to scanning attacks, which don't do anything illegal. A penetration attack can gain control of a system by exploiting a variety of software flaws. The most common flaws and the security consequences of each are explained and enumerated below.

While penetration attacks vary tremendously in details and impact, the most common types are:

User to Root: A local user on a host gains complete control of the target host

Remote to User: An attacker on the network gains access to a user account on the target host

Remote to Root: An attacker on the network gains complete control of the target host

Remote Disk Read: An attacker on the network gains the ability to read private data files on the target host without the authorization of the owner

Remote Disk Write: An attacker on the network gains the ability to write to private data files on the target host without the authorization of the owner

7.2.4. Remote vs. Local Attacks

DOS and penetration attacks come in two varieties: local and remote.

7.2.4.1. **Authorized User Attack**:

Authorized user attacks are those that start with a legitimate user account on the target system. Most authorized user attacks involve some sort of privilege escalation.

7.2.4.2. **Public User Attack**:

Public user attacks, on the other hand, are those launched without any user account or privileged access to the target system. Public user attacks are launched remotely through a network connection using only the public access granted by the target.

One typical attack strategy calls for an attacker to use a public user attack to gain initial access to a system. Then, once on the system, the attacker uses authorized user attacks to take complete control of the target.

7.2.5. Determining Attacker Location from IDS Output

In notifications of a detected attack, IDSs will often report the location of a attacker. This location is most commonly expressed as an source IP address. The reported address is simply the source address that appears in the attack packets. As attackers routinely change IP addresses in attack packets, this does not necessarily represent the true source address of the attacker.

The key to determining the significance of the reported source IP address is to classify the type of attack and then determine whether or not the attacker needs to see the reply packets sent by the victim.

If the attacker launches a one-way attack, like many flooding DOS attacks, where the attacker does not need to see any reply packets, then the attacker can label his packets with random IP addresses. The attacker is doing the real world equivalent of sending a postcard with a fake return address to fill a mailbox so that no other mail can fit into it. In this case, the attacker cannot receive any reply from the victim.

However, if the attacker needs to see the victim's replies, which is usually true with penetration attacks, then the attacker usually cannot lie about his source IP address. Using the postcard analogy, the attacker needs to know that his postcards got to the victim and therefore must usually label his postcards with his actual address.

In general, attackers must use the correct IP address when launching penetration attacks but not with DOS attacks.

However, there exists one caveat when dealing with expert attackers. An attacker can send attack packets using a fake source IP address, but arrange to wiretap the victims reply to the faked address. The attacker can do this without having access to the computer at the fake address. This manipulation of IP addressing is called "IP Spoofing."

7.2.6. IDSs and Excessive Attack Reporting

Many IDS operators are overwhelmed with the number of attacks reported by IDSs. It is simply impossible for an operator to investigate the hundreds or even thousands of attacks that are reported daily by some IDSs. The underlying problem is not in the number of attacks, but how IDSs report those attacks.

Some IDSs report a separate attack each time an attacker accesses a different host. Thus, an attacker scanning a subnet of a thousand hosts could trigger a thousand attack reports. Some vendors have proposed a solution to this problem. Their newest IDSs are beginning to effectively combine redundant entries and to present to the operator those attacks of highest importance first.

7.2.6.1. Attack Naming Conventions

Until recently, there was no common naming convention for computer attacks or vulnerabilities. This made it very difficult to compare the effectiveness of different IDSs as each vendor's IDS generated a different list of results when analyzing events reflecting the same set of attacks. This also made it difficult to coordinate the use of more than one type of IDS in a network, as different IDSs would generate different messages when they detected the same attack.

Fortunately, there are efforts underway within the network security community to devise a common nomenclature for computer vulnerabilities and attacks. The most popular of these is the Common Vulnerabilities and Exposures List (CVE) and is maintained by MITRE with input from a variety of security professionals worldwide. Many network security product vendors have agreed to make their products CVE-compatible. The CVE list can be searched and viewed using NIST's ICAT vulnerability index: http://icat.nist.gov/. The main CVE web site is: http://cve.mitre.org.

7.2.6.2. Attack Severity Levels

Many IDSs assign a severity level to detected attacks. They do this to help IDS operators accurately assess the impact of an attack, so that appropriate actions can be taken. However, the impact and severity of an attack are highly subjective, and are not necessarily one and the same, depending upon the target network and environment of the organization that hosts that network. For example, if an attacker launches a highly effective Unix attack against a large heterogeneous network, the impact of the attack for a network segment that is exclusively Windows-based may be low, while the impact of the attack on the entire network (and thus the severity of the attack) remains high. Thus, the severity levels reported by IDSs are useful information for security managers, but must be considered in the context of the specific system environment in which the IDS is running.

7.3. Types of Computer Vulnerabilities

Many IDSs provide a description of the attacks that they detect, which will often include the type of vulnerability that the attack is exploiting. This information is extremely useful after an attack has occurred so that a systems administrator can research and correct the exploited vulnerability. NIST recommends the use of the ICAT Metabase project for researching and fixing vulnerabilities in organization’s networks. ICAT will give readers thousands of examples of real world computer vulnerabilities with links to detailed descriptions and fix information. ICAT is available at http://icat.nist.gov.

In this section, we will discuss the major types of vulnerabilities. Many different schemes have been proposed to classify vulnerabilities and we shall not enumerate them here. However, some standard terminology has developed. Below is a list of some of the more common vulnerability types:

7.3.1. **Input Validation Error**:

In an input validation error, the input received by a system is not properly checked, resulting in a vulnerability that can be exploited by sending a certain input sequence. There are two important types of input validation errors: buffer overflow and boundary condition errors.

7.3.1.1. Buffer Overflow (subset of input validation errors):

In a buffer overflow, the input received by a system is longer than the expected input length but the system does not check for this condition. The input buffer fills up and overflows the memory allocated for the input. By cleverly constructing this extra input, an attacker can cause the system to execute instructions on behalf of the attacker. An example of a buffer overflow vulnerability is the fingerd exploit, in which an attacker sends a Unix finger command to a system, with an argument that is longer than the 80(?) characters allocated.

7.3.1.2. Boundary Condition Error (subset of input validation errors):

In a boundary condition error, the input being received by a system, be it human or machine generated, causes the system to exceed an assumed boundary. The overrun thereby represents a vulnerability. For example, the system may run out of memory, disk space, or network bandwidth. Another example is that a variable might reach its maximum value and roll over to its minimum value. Yet another example is that the variables in an equation might be set such that a division by zero error occurs. A boundary condition error is a subset of the class of input validation errors. While it could be argued that buffer overflow is a type of boundary condition error, we put buffer overflow in a distinct category given its commonality and importance.

7.3.2. Access Validation Error:

In an access validation error, the system is vulnerable because the access control mechanism is faulty. The problem lies not with the user controllable configuration of the access control mechanism but with the mechanism itself.

7.3.3. Exceptional Condition Handling Error:

In an exceptional condition handling error, the system somehow becomes vulnerable due to an exceptional condition that has arisen. The handling (or mishandling) of the exception by the system enables a vulnerability.

7.3.4. Environmental Error:

In an environmental error, the environment in which a system is installed somehow causes the system to be vulnerable. This may be due, for example, to an unexpected interaction between an application and the operating system or between two applications on the same host. Such a vulnerable system may be perfectly configured and provably secure in the developers test environment, but the installation environment somehow violates the developer’s security assumptions.

7.3.5. Configuration Error:

A configuration error occurs when user controllable settings in a system are set such that the system is vulnerable. This vulnerability is not due to how the system was designed but on how the end user configures the system. We consider it a configuration error when a system ships from a developer with a weak configuration.

7.3.6. Race Condition:

Race conditions occur when there is a delay between the time when a system checks to see if an operation is allowed by the security model and the time when the system actually performs the operation. The real problem is when the environment changes between the time the security check is performed and when the operation is performed, such that the security model no longer allows the operation. Attackers take advantage of this small window of opportunity and convince systems to perform illegal operations like writing to the password file while in the high-privilege state.

8. The Future of IDSs

Although the system audit function that represents the original vision of IDSs has been a formal discipline for almost fifty years, the IDS research field is still young, with most research dating to the 1980s and 1990s. Furthermore, the wide-scale commercial use of IDSs did not start until the mid-1990s.

However, the Intrusion Detection and Vulnerability Assessment market has grown into a significant commercial presence. Technology market analysts predict continued growth in the demand for IDS and other network security products and services for the foreseeable future (with IDS product sales projected to reach $978 million by 2003.) [5]

Even while the IDS research field is maturing, commercial IDSs are still in their formative years. Some commercial IDSs have received negative publicity due to their large number of false alarms, awkward control and reporting interfaces, overwhelming numbers of attack reports, lack of scalability, and lack of integration with enterprise network management systems. However, the strong commercial demand for IDSs will increase the likelihood that these problems will be successfully addressed in the near future.

We anticipate that the improvement over time in quality of performance of IDS products will likely parallel that of anti-virus software. Early anti-virus software created false alarms on many normal user actions and did not detect all known viruses. However, over the past decade, anti-virus software has progressed to its current state, in which it is transparent to users, yet so effective that few doubt its effectiveness.

Furthermore, it is very likely that certain IDS capabilities will become core capabilities of network infrastructure (such as routers, bridges and switches) and operating systems. In this case, the IDS product market will be able to better focus its attention on resolving some of the pressing issues associated with the scalability and manageability of IDS products.

There are other trends in computing that we believe will affect the form and function of IDS products including the move to appliance-based IDSs. It is also likely that certain IDS pattern-matching capabilities will move to hardware in order to increase bandwidth. Finally, the entry of insurance and other classic commercial risk management measures to the network security arena will drive enhanced IDS requirements for investigative support and features.

9. Conclusion

IDSs are here to stay, with billion dollar firms supporting the development of commercial security products and driving hundreds of millions in annual sales. However, they remain difficult to configure and operate and often can’t be effectively used by the very novice security personnel who need to benefit from them most. Due to the nationwide shortage of experienced security experts, many novices are assigned to deal with the IDSs that protect our nation’s computer systems and networks. Our intention, in writing this document, is to help those who would take on this task.

We hope that this publication, in providing actionable information and advice on the topics, serves to acquaint novices with the world of IDSs and computer attacks. The information provided in this bulletin is by no means complete and we recommend further reading and formal training before one takes on the task of configuring and using an intrusion detection system.

[5] International Data Corporation, 2000.

Appendix A – Frequently Asked Questions about IDSs

1. **What is intrusion detection?**

   Intrusion detection is the process of monitoring the events occurring in a computer system or network and analyzing them for signs of *intrusions,* defined as attempts to compromise the confidentiality, integrity, or availability of a computer or network.

2. **What is an intrusion detection system (IDS)?**

   An intrusion detection system is a software or hardware device that automates the intrusion detection process.

3. **How do IDSs work?**

   Intrusion detection systems are made up of three functional components, information sources, analysis, and response. The system obtains event information from one or more information sources, performs a pre-configured analysis of the event data, and then generates specified responses, ranging from reports to active intervention when intrusions are detected.

4. **Why should I use an IDS, especially when I already have firewalls, anti virus tools, and other security protections on my system?**

   Each security protection serves to address a particular security threat to your system. Furthermore, each security protection has weak and strong points. Only by combining them (this combination is sometimes called *security in depth*) do you protect from a realistic range of security attacks.

   Firewalls serve as barrier mechanisms, barring entry to some kinds of network traffic and allowing others, based on a firewall policy. IDSs serve as monitoring mechanisms, watching activities, and making decisions about whether the observed events are suspicious. They can spot attackers circumventing firewalls and report them to system administrators, who can take steps to prevent damage.

5. **What are the different types of IDSs?**

   There are many ways of describing IDSs. The primary descriptors are the system monitoring approaches, the analysis strategy and the timing of information sources and analysis. The system monitoring approaches are network-based, host-based, and applications-based. The analysis strategies are misuse detection and anomaly detection. The timing categories are interval-based (or batch mode) and real-time. The most common commercial IDSs are real-time network-based systems.

6. **How do I select the best IDS for my organization?**

   The best IDS for your organization is the IDS that best satisfies the security goals and objectives of your organization, given the constraints of the organization. Governing factors are usually defined as the following:

   - System environment, in terms of hardware and software architectures.
   - Security environment, in terms of policy, existing security mechanisms, and constraints.

- Organizational goals, in terms of functional goals of the enterprise (for instance, e-commerce organizations might have different goals and constraints from manufacturing organizations.)
- Resource constraints, in terms of acquisition, staffing, and infrastructure.

7. **What are the limitations of IDSs?**

IDSs have many limitations. These are the major ones:

- They aren't scalable to large or distributed enterprise networks.
- They can be difficult to manage, with awkward user control and alarm display interfaces
- Different commercial IDSs rarely interoperate with each other, so you may not be able to consolidate your IDSs across your enterprise if you use more than one vendor's IDS.
- Commercial IDSs rarely interoperate with other security or network management packages
- There are significant error rates, especially false positives, in IDS results. These can take up a great deal of a security staff's time and resource.
- They cannot compensate for significant deficiencies in your organizations security strategy, policy, or security architecture.
- They cannot compensate for security weaknesses in network protocols
- They cannot substitute for other types of security mechanisms (such as Identification and Authentication, encryption, single sign on, firewalls, or access control)
- They cannot, by themselves, completely protect a system from all security threats

8. **What is the difference between vulnerability analysis systems and intrusion detection systems?**

Vulnerability analysis systems are very similar to intrusion detection systems, as they both look for specific symptoms of intrusions and other security policy violations. However, vulnerability analysis systems take a *static* view of such symptoms, whereas intrusion detections look at them *dynamically*. This is the difference between taking a snapshot of an incident versus video taping it.

9. **How can the two systems interact?**

Consider this analogy: Intrusion Detection Systems are analogous to security monitoring cameras. Standard IDSs perform real-time continuous monitoring and analysis of event data; hence they are analogous to cameras that record video images. Vulnerability assessment systems perform interval-based monitoring and analysis of system state; hence they are analogous to cameras that take snapshots. Vulnerability assessment systems can determine whether there is a problem at a particular point in time, and can furthermore make this determination for a modest investment of time and processing load. IDSs can, on the other hand, tell very reliably whether there are problems over a time interval, and furthermore tell the conditions that enabled the problem, as well as the damage caused by the problem.

Appendix B - IDS resources

Acquiring, deploying, and maintaining an IDS is a complex task. Fortunately, many excellent resources in the form of books and seminars exist to guide the public on IDS technology. Several free IDS resources are available:

1. For an overview of IDSs and their capabilities, read the white paper "An Introduction to Intrusion Detection Assessment for System and Network Security Management" at http://www.icsa.net/services/consortia/intrusion/intrusion.pdf.
2. For a survey of commercially available IDSs that allows one to easily compare features, read the "Intrusion Detection System Product Survey" published by the Los Alamos National Laboratory and found at http://lib-www.lanl.gov/la-pubs/00416750.pdf.
3. NIST's ICAT vulnerability index allows you to search for information about specific vulnerabilities. It is found at: http://csrc.nist.gov/icat.
4. Information on the computer attacks that IDSs detect can be found in the May 1999 *ITL Bulletin* entitled "Computer Attacks: What They Are and How to Defend Against Them", available at http://www.nist.gov/itl/lab/bulletns/cslbull1.htm.
5. Snort is a lightweight network intrusion detection system, which can perform a variety of traffic logging and analysis functions on IP networks. It is a freeware product, available under the terms of the GNU General Public License as published by the Free Software Foundation. Snort has an extensive database of over a thousand attack signatures. Both Snort and the attack signature database are found at http://www.snort.org.
6. The Center for Education and Research in Information Assurance and Security (CERIAS) at Purdue University, has produced many widely-used network security tools, including the first widely used vulnerability assessment tool, COPS, and the first widely-used file integrity checker, Tripwire. CERIAS has an extensive ftp repository of freeware tools for security managers, including many intrusion detection and vulnerability assessment tools. All are found at http://www.cerias.purdue.edu.
7. SecurityFocus.com has a web-accessible reference site for intrusion detection, that features news, information, discussions, and tools. It is found at http://www.securityfocus.com
8. Talisker's Network Security Tool Site has an extensive, well-maintained reference library of commercial IDS products. It is found at http://www.networkintrusion.co.uk.
9. There are several books available on intrusion detection, including:
   - Bace, Rebecca G., *Intrusion Detection,* Macmillan Technical Publishing, 2000.
   - Amoroso, Edward G., *Intrusion Detection: An Introduction to Internet Surveillance, Correlation, Trace Back, Traps, and Response,* Intrusion.net, 1999.
   - Northcutt, Stephen, *Network Intrusion Detection: An Analyst's Handbook,* New Riders, 1999.

NOTE: Any mention of commercial products is for information only; it does not imply recommendation or endorsement by the National Institute of Standards and Technology nor does it imply that the products mentioned are necessarily the best available for the purpose.