IDS Introduction

IDS Implementations

To detect and prevent attacks, IDS solutions are implemented by one of two methods: profiles or signatures. The following two sections discuss these two methods, including their advantages and limitations.

Profiles

Profile-based systems look for anomalies in traffic. They do this first by capturing traffic under normal circumstances, called a profile. They then use the profile to compare it to traffic. A profile contains information about traffic patterns and statistics. Any traffic that does not fit the profile is considered an anomaly and could be a network attack. Because of this process, profile-based systems often are called anomaly-detection systems.

One of the main advantages that profile-based systems have is that they have a better likelihood of detecting new kinds of attacks over signature-based systems. For example, assume that a hacker discovers a security hole in a web server and tries to access a special file called access.htm that has been left there by the default installation. This file is never referenced, but it provides back-door management access to the web server. Normally, this should have been removed, but some administrators might have forgotten to do this. This file is never accessed in normal operations, so when a hacker tries to access this file, a profile-based system should raise an alarm because this type of activity is considered an anomaly.

Profile-based systems have some of issues with detecting attacks, however:

- They are prone to a higher level of false readings than signature-based systems because traffic patterns vary. This can be a problem when introducing new applications and devices in the network. Plus, it takes more time to determine whether an attack really occurred or whether it was a false reading, increasing your management overhead.
- Because of changing traffic patterns and network topologies, a lot of time must be spent recreating a current profile, increasing your management.
- When creating the actual profile to compare to traffic, if the captured profile includes an attack, the attack is treated as normal traffic. Therefore, if the attack occurs again and is compared to the profile that contains the attack, the IDS profile device considers this normal and does not raise an alert. Therefore, a lot of care needs to be taken when capturing traffic profiles.

Signatures

Signature-based systems compare traffic to signatures to determine whether an attack is occurring. A signature is basically a grouping of matching criteria (commonly referred to as a template) that the IDS solution should use when determining whether an attack is occurring. Incoming traffic is compared to a list of signatures; if there is a match, an alarm is raised. This type of system is commonly called misuse detection.

Unlike profile-based systems, signature systems have fewer false readings because they are looking for very specific things in traffic. As an example, if you know the mechanics of a web server access attack, such as the information contained in a malformed URL, a signature-based system would look for this specific information in HTTP segments.

Signature-based solutions have two main drawbacks, however:

- They can detect only attacks that have been programmed by their installed signatures. If a new attack is discovered, there is a high likelihood that a signature-based solution will not detect the attack. Therefore, you must ensure that your IDS signature solution always has the most up-to-date signatures installed.
- Signature-based solutions have problems when dealing with event horizon attacks, which are attacks that occur over a period of time. A very good hacker might plan reconnaissance and access attacks over a long period of time, making them more difficult to detect: Imagine a ping or port-scan sweep that occurs over a period of hours or days. With a profile-based system, this probably would be flagged as an anomaly. However, a signature-based solution likely would not be capable of detecting this attack because it could not buffer up all of the traffic over a long period of time.

Complications with IDS Systems

As in the last example of dealing with attacks spanning a long period of time, when an IDS system detects a ping sweep or port scan reconnaissance attack, it must look at dozens, if not hundreds, of packets when determining whether an attack is occurring. Because of the limitations of both approaches, an IDS solution might do the following:

- Indicate that an attack is occurring, when there really isn't an attack
- Miss an attack by a hacker because the attack occurred over a long period of time or because the hacker carefully hid the attack among normal traffic patterns

In other words, IDS is not an exact science. Many changes have taken place in IDS solutions in the last few years, and a lot of work still has to be done to improve their effectiveness. Therefore, you should not rely solely on an IDS solution for all your security needs. Instead, an IDS solution should be one part of your security solution and should be complemented by other technologies, such firewalls and VPNs.

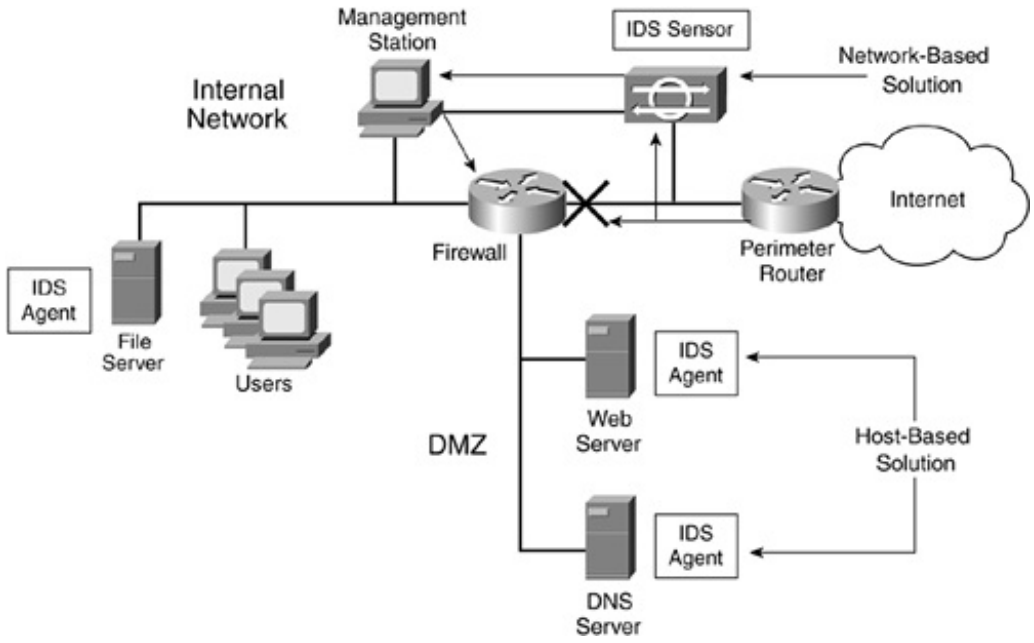
NOTE

Some products are a hybrid implementation, containing both signature- and profile-based implementations.

IDS Solutions

Two types of IDS designs exist: network- and host-based solutions. Figure 16-1 shows an example of the two solutions. The next two sections cover these solutions in more depth.

Figure 16-1. Network- and Host-Based Solutions



Network-Based Solutions

A network-based solution uses a device called a sensor to monitor traffic on a segment. The sensor examines packets and their contents and determines whether an attack is occurring. Basically, a sensor is an enhanced protocol analyzer: It captures packets that it sees and then compares these packets to the sensor's internal intrusion-detection rules. Cisco offers the following network-based solutions:

- 4200 series hardware sensors
- IDS modules for the Catalyst 6500 switches
- Network Module Cisco IDS (NM-CIDS), which is basically a 4200 sensor in a network module for a Cisco IOS router
- Routers with the Cisco IOS Firewall feature set
- PIX firewalls

Two design philosophies exist concerning network-based IDS solutions: inline and passive monitoring. An inline system has the actual packet traffic pass through the IDS device. One advantage that this design offers is that the IDS device can take a more proactive role in dealing with attacks; however, its main disadvantage is that it tries to perform IDS while also passing traffic between interfaces at wire speeds. Examples of inline designs include the Cisco IOS router and the PIX IDS. Cisco is working on other products in this category.

In a passive-monitoring design, the sensor has an interface attached to the segment that it wants to monitor. This can be accomplished by using a network tap, a hub, or a switched-port analyzer (SPAN) connection on a switch. One advantage of this approach is that the monitoring interface is passive and only receives traffic (only in rare cases will it generate traffic); therefore, it is much less susceptible to an attack than an inline solution is. In some implementations, the sensor itself can take an appropriate action when an attack occurs; in other implementations, a central management solution handles the action. However, the disadvantage of this approach is that the sensor is reacting to the attack after the packets already might have reached the victim. Examples of passive-monitoring IDS devices include the 4200 sensors and the IDS Module for the Catalyst 6500 switches.

A sensor can take these three main actions:

- Log the action or attack (alarm)
- Block the offending traffic (drop)
- Terminate an offending TCP connection (reset)

One of the main advantages of a network-based IDS solution is that it centralizes your IDS process and reduces the number of IDS devices that you need. You put them at critical points in your network, such as your network perimeter and your internal backbone, to monitor traffic.

CAUTION

When choosing a network-based IDS solution, it is important to pick a solution that can handle the analysis of packets at wire speeds. For example, if your sensor is monitoring a 100-Mbps link, but it can process only 50 Mbps of traffic, in some situations, the sensor will drop and miss packets. Therefore, it might miss an attack.

Host-Based Solutions

In a host-based IDS solution, agent software is running on a host system, such as a server or PC. These solutions come with features that include the following:

- Most of these solutions check to make sure that key application and operating system files are not altered.
- Most of these solutions check for unauthorized access.
- Most of these solutions check for appropriate use of the application.
- Some of these solutions check for reconnaissance and DoS attacks.
- All of these solutions create log files, which can be kept locally or forwarded to a central repository (a good centralized solution is important in large IDS deployments).

Host-Based Versus Network-Based

One of the advantages of a host-based system is that it has to handle only traffic sent to the device that it is running on. Sometimes a network-based solution has problems with handling a lot of traffic, and it can be difficult to manage when you need to define per-host IDS policies. However, the main disadvantage of host-based systems is that the more you have, the more difficult they are to manage.

Therefore, a good enterprise solution typically has a mix of both host- and network-based IDS solutions. Sensors are used at the perimeter and backbone of the network, as well as other key access points, such as dialup. This provides a broad range of coverage for the entire network. Agent software is then installed on key hosts to provide extra protection for applications running on the host.

IDS Concerns

To be effective, IDS solutions must be capable of detecting and responding to network and host threats in a real-time manner. However, not every IDS solution fits into every network. This section focuses on some concerns about IDS solutions.

Installed Components

IDS network (sensor) solutions typically have the following components:

- CPU and memory
- Disk space
- Two or more NICs

A hardware sensor is typically a standalone device: It has its own chassis, CPU, RAM, and operating system. The CPU and RAM are critical in an IDS solution because the system might have to process millions of packets per second.

Many popular IDS solutions run on Linux or some variety of UNIX; others run on Microsoft Windows and even proprietary operating systems. For example, the newest Cisco software for its 4200 series sensors runs on top of Linux. A disk drive typically is required to store binary files and logging information. One concern with using a commercial operating system is that it is more susceptible to network attacks than a proprietary operating system; extreme care must be taken in securing the sensor itself, which includes the operating system that it uses.

Because of security risks, many IDS solutions have two NICs (interfaces): One monitors traffic, and one sends logging and reporting information to a central repository. This allows for out-of-band management. You should ensure that the management interface is protected. The sensing interface should not have an IP address; instead, it should be a promiscuous interface that only senses traffic. Some sensors support multiple NICs to monitor multiple segments. One concern with multiple sensing interfaces is the capability of the sensor to process all of its monitored traffic at wire speeds.

Detecting Intrusions

The main purpose of an IDS solution is to detect intrusions. It accomplishes this by examining packets, reassembling fragmented packets into whole packets, and matching these against an installed base of signatures or profiles. The last section mentioned the concern of the sensor being capable of keeping up with the amount of traffic that it needs to process. Another concern is that it have the capability to store packets for a period of time to detect different kinds of attacks, such as those that use fragments. For an IDS solution to detect certain kinds of fragment attacks, it must be capable of storing the fragments and, when it has all the fragments, reassembling them. In large-scale fragment attacks, this requires a lot of memory in the sensor.

Responding to Intrusions

When an attack is detected, an IDS solution should take some kind of action, which can include the following:

- Generate an alarm
- Terminate an offending TCP session
- Drop or block traffic from the offender

All IDS solutions can perform the first action. Depending on the sensor, alarm messages are stored locally, on a management console, or both. An alarm can be construed as a log message, but it differs in the following respects:

- It contains information about the severity of the attack and recommends solutions for dealing with it.
- It can be something as simple as a log message, but it also can take on other means, such as an e-mail, a page, or some other means of communication.

For TCP-based attacks, some sensors can tear down the TCP connection when an attack is detected. This feature commonly is called TCP reset. Basically, the sensor sends TCP segments to the victim, closing the session. With Cisco IDS sensors, such as the 4200s, the sensor sends a TCP reset to both the attacker and the victim by spoofing the source and destination addresses of the original packet. Normally, the TCP reset method is coupled with blocking (sometimes called shunning). Blocking blocks all traffic from the offender (or the entire network of the offender).

It is important to understand the capabilities of a sensor when making a purchasing decision. One problem with sensors that only generate logs or alarms is that you manually must react to the threat. Sensors that support the TCP reset and shunning or blocking feature enable you to take a more proactive approach when dealing with security threats and attacks.