# Intrusion Prevention Systems: Security's Silver Bullet?

**Dinesh Sequeira**

## Next-generation products don't just detect attacks, they try to stop them. Here's how they work.

Traditionally, firewalls and anti-virus programs try to block attacks, and intrusion detection systems (IDSs) identify attacks as they occur. Such techniques are crucial to network security, but have limitations. A firewall can stop attacks by blocking certain port numbers, but it does little to analyze traffic that uses allowed port numbers. IDSs can monitor and analyze traffic that passes through open ports, but do not prevent attacks.

With the proliferation of sophisticated attacks and the discovery of new vulnerabilities, new methods are needed to protect precious data and network resources. Intrusion prevention systems (IPSs) use new proactive approaches that block attacks before damage is done.

### Intrusion Detection Systems (IDSs)

To understand this new type of system, let's start by differentiating it from intrusion detection. IDSs identify the presence of malicious code within traffic that flows through the holes punched into the firewall, our first layer of defense.

However, the term "intrusion detection" is a bit of a misnomer. Richard Kemmerer and Giovanni Vigna of the University Of California, Santa Barbara, explain in an article in the *IEEE Security and Privacy* magazine: "Intrusion detection systems do not detect intrusions at all—they only identify evidence of intrusion, either while in progress or after the fact."

An IDS identifies security threats by detecting scans, probes and attacks, but does not block these patterns; it merely reports that they took place. Nevertheless, IDS logged data is invaluable as evidence for forensics and incident handling. IDSs also detect internal attacks, which are not seen by the firewall, and they aid in firewall audits.

IDSs can be divided into two main categories, based on the IDS alarm triggering mechanism: anomaly detection-based IDS and misuse detection-based IDS.

Anomaly detection based IDSs report deviations from "normal" or expected behavior. Behavior other than "normal" is considered an attack and is flagged and recorded. Anomaly detection is also referred to as profile-based detection. The profile defines a baseline for normal user tasks, and the quality of these user profiles directly affects the detection capability of the IDS. Techniques for constructing user profiles include:

- **Rule-based approach—**Normal user behavior is represented by creating rules, but analyzing normal traffic is a complicated task. A related approach is protocol anomaly detection (see below for further explanation).
- **Neural networks—**These systems are trained by presenting them with a large amount of data, along with rules about data relationships. They then determine if traffic is normal or not; abnormal traffic raises an alarm.
- **Statistical approach—**Activity profiles characterize the behavior of system or user traffic. Any deviation from normal triggers an alarm.

The advantage of anomaly detection is that it can detect previously unknown attacks and insider attacks, without the need for "signatures"—i.e., predefined attack profiles. Another benefit of anomaly detection is that it's impossible for the attacker to know what activity generates an alarm, so they cannot assume that any particular action will go undetected.

The disadvantage of anomaly detection is that it generates a large number of "false positives"—i.e., alerts that are produced by legitimate activity. Furthermore, besides being complicated and hard to understand, building and updating profiles also require a lot of work.

The other main approach, *misuse-detection based IDS* (also called signature-based IDS), triggers an alarm when a match is found to a "fingerprint"—a signature contained in a signature database. These "fingerprints" are based on a set of rules that match typical patterns of exploits used by attackers. Since there is a known database of exploits, there are few false positives.

The disadvantage is that misuse-detection IDSs can only detect already-known attacks. Besides, the "fingerprints" database needs to be continuously updated to keep up with new attacks. Most

*Dinesh Sequeira is an independent network consultant specializing in network security and wireless networks. He is also involved in training as SANS Institute's Online Mentor for Atlanta. He can be reached at dsequeir@spsu.edu.*

IDS products in the market today use misuse detection.

Another way to classify IDSs is by monitoring location:

- *Network-based IDSs (NIDS)* sit behind the firewall, on the demilitarized zone (DMZ) or the private network, and sniff packets on a network segment in promiscuous mode (see "Seven Key Definitions"), invisible to the attacker. NIDS monitors and analyzes network traffic and can use either anomaly- or misuse-detection techniques.

While the firewall screens out unwanted traffic, the NIDS will alert the security manager to what is "leaking" through the firewall. Based on data collected, future attacks can be prevented or the attack can be contained.

## Cisco Moves In On Intrusion

Once the omnivore of the networking industry, Cisco has put itself on a major diet when it comes to acquiring smaller companies. That's why it's notable that the vendor's two most recent purchases have been of companies that created intrusion prevention technology—or "intrusion protection," in Cisco-speak.

Last October, Cisco announced the purchase of Psionic Software of Austin, TX, and in January, the networking giant announced it was buying Okena. Psion's specialty was systems aimed at eliminating the serious problem of "false positives" in intrusion detection/prevention, while Okena created a host-based product called StormWatch (see the main article for details on these concepts).

Cisco rolled these acquisitions together with its own technology and came up with a new suite of security capabilities, that will be incorporated into a new release of Cisco's intrusion detection software (version 4.0).

According to Joel McFarland, manager of security platforms at Cisco's VPN and security business unit, the newest announcement also includes the second generation of IDS capabilities for a module that runs in Cisco's Catalyst 6500 switch (see *BCR*, October 2002, pp. 62–63). Finally, Cisco announced the latest in its series of IDS appliances, the IDS 4250-XL, which supports 1-Gbps line rates.

### Weeding Out Alarms

Cisco incorporated the Psionic technology into an element it calls the Intelligent Investigation Element, which the vendor claims can reduce the false positive rate by 80–95 percent. This is accomplished through "just-in-time analysis," as McFarland called it.

As an example of how this works, McFarland posited that an IDS sensor in a Catalyst switch determines that someone is trying to use the IIS Unicode attack—which only affects Windows NT—against a Linux system. The Intelligent Investigation Element "will immediately investigate the target IP address of that attack and say, Are you a Linux box? Are you a Windows box?" McFarland said. Upon determining the nature of the threat, the investigation element will issue an alert—low if the target is Linux, high if the target had been a Windows NT server.

"What we're focusing on here is reducing the operational burden of the user to manually investigate this particular attack," McFarland said. "We weed out the information that's not applicable to the network."

### All The Way To The Desktop

The value of having a capability like Okena's has already been validated by Cisco's customers over the past 18 months, according to Joel McFarland. Ironically, that validation came when Cisco was OEMing solutions from Entercept, a key Okena competitor.

McFarland didn't say why Cisco chose Okena's solution over Entercept's, but he did say Cisco is emphasizing greater endpoint security capabilities. "Cisco is taking a stronger position in endpoint security as a critical component of providing complete solutions to our customer," McFarland said. "We believe there's a lot of harmony and collaboration between the endpoint systems and the network security systems, which together is the solution that our customers are asking us to solve."

### Conclusion

According to Joel McFarland, a successful intrusion protection system will combine such approaches as those described in the accompanying article. "We use a kind of hybrid system of identifying bad stuff," he said. "We use multiple different methods—stateful pattern matching, we look into the protocols using protocol decode analysis. We're looking at traffic-based anomaly activities as well as protocol anomaly detection."

The key is to offer a system that responds to the ever-evolving security threat. "The argument two years ago, when you compared one company's IDS system to another company's, used to be: Hey, I have more signatures than you, and therefore I'm better," McFarland went on. "Recently it's become a debate that: I use more methods to identify bad things than you, therefore I'm implying that I'm better than you."□

*—Eric Krapf*

**IDSs are becoming more sophisticated, but so are evasion techniques**

NIDSs are compute-intensive: They need to keep up with the high volume of network traffic, or else they could miss attacks. High speed is also essential for low latency. Thus, this type of product is usually available as dedicated hardware appliances.

■ *Host-based IDS (HIDS)* software is run on each host. The software monitors log data and compares it to attack signatures to detect intrusions. The applications' interaction with the host operating system is also monitored for any intrusive activity.

HIDS take a closer and deeper look at the activity of attack tools on the host, which cannot be done with a NIDS. HIDS are generally employed on Web and DNS servers, as well as other hosts that make a prime target for attackers. Tripwire, a file integrity-checking tool, is a classic example of HIDS, generating alerts when changes to a file are detected.

### IDS Evasion Techniques

Although there are various categories of intrusion detection systems, evasion techniques have also become sophisticated. The basic idea behind evasion is to fool the IDS into seeing different data than what the target host will see, thus allowing the attacker to slip through undetected. Some IDS evasion techniques are:

■ **Polymorphic buffer-overflow attacks—** These alter the attack's shell code. One example is ADMutate, an online tool that can take an attack's shell code and transform it in such a way that the code looks different from the known attack signature but is functionally equivalent. Once the attack gets to the target, it reassembles, having eluded the intrusion detection system.

■ **Path obfuscation—**An attacker can use a Web browser's URL to enter a path statement in order to access a file on the Web server with the intention of causing damage, or to retrieve sensitive information.

Normally, the path statement would be incorporated into the attack signature, and the attack could be recognized. However, the attacker could alter the URL's path statement to appear different to evade detection and cause harm. For example, "/winnt/. /. /. /test" is the same as "/winnt/test," but the signatures don't match, and so an IDS trained to alert on "/winnt/test" will miss this attack.

■ **Hex encoding—**Hex encoding can be used to represent characters in URLs. For example, "%20" means "hex 20," and is the equivalent of a single space in ASCII. The HTTP protocol uses hex encoding, but not all IDSs understand it and so could miss an attack

■ **Unicode directory traversal—**Directory traversal exploits use strings like ".. /.. /.. / ". Most IDSs have signatures to detect this, but attackers replace the " / " with the Unicode equivalent, "%c0%af," and evade the IDS and thus traverse other directories. In fact, many variations of the same string could be created.

■ **Protocol anomalies—**Host applications can vary in protocol implementation, as RFCs may not be accurately specified. Network IDS may have a different interpretation from that on the host, and attackers can exploit this discrepancy.

■ **Fragmentation—**Fragmented packets are reassembled only at the destination. Fragmented pieces of attack code could slip through the network undetected, unleashing their evil intent when they reach the end host.

## Seven Key Definitions

Among the key concepts related to intrusion detection and prevention, seven may require more explanation:

*1.) What is a false positive?*A false positive is an alert sent out when an IDS sensor misinterprets one or more legitimate, benign packets as an attack.

*2.) What is a false negative?* A false negative occurs when an IDS sensor misinterprets one or more malicious packets as being harmless.

*3.) What is a buffer-overflow exploit?* Application programs have a fixed-size buffer that holds data. If an attacker sends too much data and the application program has not been written to check the size of the data (which is a bug in the application software!), the buffer overflows. The server may then execute the data that "overflowed" as if it were a program. This "overflowed" data is the attacker's malicious program. The server may then execute this malicious code, which has privileged access for executing commands, altering the system configuration, installing Trojan horses or back doors.

*4.) What is a sandbox?* A sandbox is an area on the host that has restricted access to the rest of the system resources. The sandbox software is a virtual machine, which executes code in isolation from the operating system. Users' software applications can play in the sandbox, but can't do anything destructive.

*5.) What is "promiscuous" mode?* When a network interface card (NIC) is set up to read all network traffic, even those that are not addressed to it, it is said to be in "promiscuous" mode.

*6.) What is "normalization?"* Normalization is the process of removing exploitable ambiguities in network traffic before the traffic is evaluated for malicious code, thus removing evasion opportunities.

*7.) What is SQL Injection?* SQL Injection is a method used to attack, alter or retrieve data in a database through a Web-based application□

### Intrusion Prevention Systems

Traditional signature-based IDSs focus on how an attack works, i.e., trying to detect certain strings. But if an attacker uses any of the IDS evasion techniques discussed above, the previously written IDS signatures no longer detect it.

In contrast, IPS focuses on what an attack does—its behavior, which does not change. In addition to using signatures, IPSs use a set of rules to represent either permissible or harmful behavior. Traffic in real time is then compared to the set of rules and either permitted or blocked.

Prevention methods implemented via IPS stop malicious behavior before it can cause any harm. Some of the common types of malicious behavior are:

- **Alteration to System Resources—**Trojan horses, root kits and back doors alter system resources like libraries, files/directories, registry settings and user accounts. By preventing alteration to system resources, hacking tools cannot be installed.
- **Privilege-Escalation Exploits—**Privilege escalation attacks try to give ordinary users root or administrator privileges. Disallowing access to resources that alter privilege levels can block exploits like Trojan horses, root kits and back doors.
- **Buffer-Overflow Exploits—**Since the exploit code invokes at least one system call, a check of whether the system call about to be executed by the operating system came from a normal application or an overflowed buffer exploit helps prevent these attacks.
- **Access To Email Contact List—**Many worms spread by mailing a copy to those in the user's Outlook contact list. Prohibiting email attachments from accessing Outlook's contact list prevents spread of these worms.
- **Directory Traversal—**The directory traversal vulnerability in different Web servers allows the hacker to access files outside the range of what the Web server would normally need to access. Preventing the hacker access to the Web server files outside its normal range can prevent such malicious activities.

### Are You Hip To HIPS?

As with IDS, there are several different approaches to implementing intrusion prevention systems. They include:

**Host-based Intrusion Prevention Systems (HIPSs)**, which protect desktops or servers by protecting the operating system from attacks like buffer overflow exploits. Products also are available to protect specific servers like Web or database servers against application attacks like Unicode directory traversal vulnerabilities (mentioned above) or SQL injection. Some of the HIPS prevention approaches are:

- *Software-based heuristics*—This approach is similar to IDS anomaly detection using neural networks (mentioned earlier) to act against new or unknown types of intrusions. HIPS add the ability to block the attacks.
- *Sandbox approach*—Mobile code like ActiveX, Java applets and various scripting languages are quarantined in a sandbox—i.e., an area with restricted access to the rest of the system. This system then runs the suspect mobile code in the sandbox and monitors its behavior. If the code violates a predefined policy, it's stopped and prevented from executing.
- *Kernel-based protection*—The kernel controls access to system resources like memory, input/output devices and CPU. In order to use these resources, user applications send requests or system calls to the kernel, which then carries out the operation. Any exploit code will execute at least one system call to gain access to privileged resources or services.

Kernel-based HIPS prevents execution of malicious system calls. Programming errors enable exploits like buffer-overflow attacks to over-write kernel memory space and crash or take over computer systems. To prevent these types of attacks, a software agent is loaded between the user application and the kernel. The software agent intercepts system calls to the kernel, inspects them against an access control list defined by a policy and then either allows or denies access to resources.

On some IPSs, the agent checks against a database of specific attack signatures or behaviors. It also could check against a database of known good behaviors or a set of rules for a particular service. Either way, if a system call attempts to run outside its allowed zone, the agent will stop the process.

Okena's StormWatch product (Okena was recently bought by Cisco—see "Cisco Moves In On Intrusion") uses a kernel-based approach and works on servers and workstations. Policies—collections of access control rules based on acceptable behavior—are available out-of-the-box for common applications such as Microsoft SQL Server, Instant Messenger and IIS Server.

Policies control what resource is being used, what operation is being invoked, and which application is invoking it. StormWatch hooks into the kernel and intercepts system calls. It is reported to have stopped the Klez worm and the recent Slammer or Sapphire worm, among others.

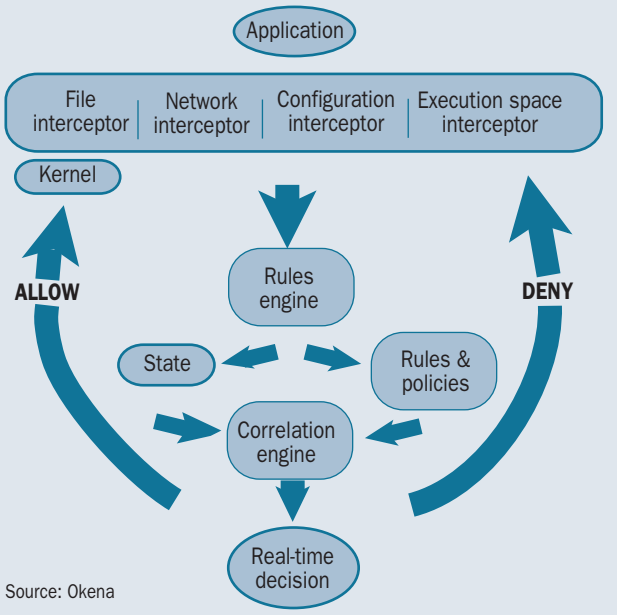
StormWatch has four interceptors (Figure 1, p. 40):

- *File System interceptor*—Intercepts all file read and write requests.
- *Network interceptor*—Intercepts packet events at the driver (NDIS) or transport (TDI) level.
- *Configuration interceptor*—Intercepts read/write requests to the registry on Windows or to rc files on Unix.
- *Execution space (run-time environment) interceptor*—Requests to write to memory not owned by the requesting application will be blocked by this interceptor. For example, buffer-overflow attacks

**IPS focuses less on how an attack looks than on what it does**

**Is a high rate of "false positives" the price of progress?**

**FIGURE 1 Okena Architecture**




Source: Okena

would be blocked here. Thus it maintains the integrity of each application's dynamic run-time.

Since StormWatch intercepts File, Network, Configuration and Run-time operations and compares them to application-specific access control rules or policies, it can track the state of an application. For example, the network interceptor provides address and port blocking like a firewall; file system and configuration interceptors monitor and prevent changes to critical files or registry keys. Network and file system interceptors provide worm prevention.

By correlating events from multiple host systems at the management station, StormWatch not only blocks the threat but also pushes out a new policy to all agents and blocks future attacks. This reduces the number of false positives and false negatives.

Storm Watch has a utility program called StormFront. It serves as a data analysis and policy creation tool, analyzing applications as they operate in a normal environment, and it generates policies. Any application behavior outside the policy would then be considered suspicious. Resources accessed by the application are separated into file, network, registry and COM categories.

Another vendor, Entercept, is a pioneer in kernel-based protection. One of its products, Entercept Standard Edition, proactively protects servers by intercepting system calls (Figure 2). Unlike Okena's StormWatch, it uses both signatures and behavior rules to stop and detect attacks. It runs only on servers; in contrast, StormWatch has server and desktop agents.

Entercept's Web Server Edition adds Web-server specific controls to the Standard Edition and protects against directory traversals (mentioned above) and remote code execution. The Database Edition adds protection against SQL injection attacks (see "Seven Key Definitions"), in addition to the Standard Edition features.

### Network IPS

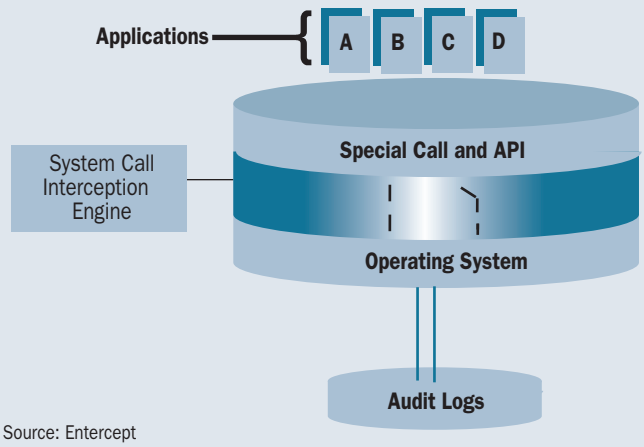
The other class of IPS, **Network-based Intrusion Prevention Systems (NIPS)** generally consist of appliance-based systems that sit inline and block suspicious traffic upon detecting an attack. They use signature detection, anomaly detection and proprietary methods to block network-level intrusions like denial-of-service attacks. All this resource-intensive processing is done with the aid of dedicated hardware boxes for speed and latency issues, and appliances are already available that work at gigabit speeds.

The disadvantage of being in-line is that if the device fails, the entire network it serves is down. This can be overcome by having failover or redundant systems, but at a cost.

Some NIPS are reported to have a high rate of false positives, but have blocked thousands of known attacks. Products are just being released and their performance needs to be evaluated, especially with new attack methods. NetScreen, TippingPoint, Top Layer Networks and IntruVert Networks are some leading NIPS vendors.

Some NIPS prevention methods are:

**FIGURE 2 Entercept Architecture**




Source: Entercept

■ **State-based signature detection—**This looks at relevant portions of traffic (which could be multiple packets) by tracking state, and based on the context specified by the user, detects attacks. It is not completely automated, as the user needs to have some prior knowledge about the attack. For example, the Love Letter worm can be detected by a rule that would read as follows: "Look for 'ILOVEYOU' in the subject field only, ignore this string anywhere else in the email."

■ **Pattern matching using regular expressions—**Some NIDS look for fixed attack patterns. A minor change like a space or tab in the attack patterns can be enough to evade detection. By using regular expression pattern-matching, NIPS can avoid missing attacks. Regular expressions provide wild-card and complex pattern matching, and are able to prevent attacks.

■ **Protocol anomaly detection—**Most NIPS vendors do detailed packet analysis and protocol decodes to ensure packets adhere to the protocol, and have no ambiguities. For example, by IP spoofing of FTP PORT commands, the attacker can tell the FTP server to open a connection to a victim's IP address and then transfer a Trojan horse to the victim. Checking for a match between the IP address in the FTP PORT command and the client's IP address can prevent this anomaly.

As mentioned earlier, fragmented packets could slip through, and when reassembled at the end host, unleash their evil intent. Normalization (see "Seven Key Definitions") can combat this tactic. It removes such ambiguities and ensures that traffic interpreted by the NIPS is the same as that seen by the end host.

■ **Traffic anomaly detection—**Attackers often conduct a port or network scan as a precursor to an attack. NIPS implement frequency and threshold triggers that alert to such scanning activity, increasing the likelihood that the attack can be prevented. Neither of the other methods mentioned above would detect this, as it has not violated any protocol or pattern.

■ **Signature detection—** This is used in conjunction with the above-mentioned methods to ward off combined attack types, which increasingly are seen on today's networks. Many IPS vendors provide or intend to provide integrated firewall/IDS/antivirus and vulnerability assessment capabilities. In addition, some IPS vendors integrate with other firewall, IDS and vulnerability assessment tools.

## Conclusion

Firewalls, antivirus, IDS and IPS have their place in the security landscape, each with its unique features, and are not competing components. Depending on its business needs, budget constraints and level of risk tolerance, the enterprise must draw up a security policy. That policy will determine the mix of components that needs to be installed to meet security goals.

Intrusion prevention is a generic marketing term. Before purchasing a product, study the detection and prevention mechanisms vendors have implemented vis-à-vis current attack methods. In general, IPS can be considered an evolution of IDS technology. Its proactive capabilities will help to keep networks safer from more sophisticated attacks.

In particular, host-based security is becoming more important today, as enterprise networks' use of tunneling and encryption puts more content out of the reach of perimeter controls such as firewalls. Even though network-based IPS will prevent attacks, some could slip through, and host-based IPS aims to prevent them. HIPS—the last line of defense—provides "operating system hardening" with greater granularity and application specific control

Bulletproof security does not exist. Security is a continuous process of monitoring, maintenance and modification. Some attacks could still slip through, and no amount of automation can replace trained and vigilant personnel. Tools like IPS can provide a silver lining if not a silver bullet!□

**There's no silver bullet; security is a process**

| Companies Mentioned In This Article |
|---|
| Cisco (www.cisco.com) |
| Entercept (www.entercept.com) |
| IntruVert Networks (www.intruvert.com) |
| Microsoft (www.microsoft.com) |
| NetScreen (www.netscreen.com) |
| Okena (www.okena.com) |
| TippingPoint (www.tippingpoint.com) |
| Top Layer Networks (www.toplayer.com) |