CHAPTER 11

# Intrusion Detection and Prevention Systems

## Introduction

Intrusion in lay terms is unwanted or unauthorized interference and as it is unwanted or unauthorized, it is normally with bad intentions. The intention of the intrusion is to collect information related to the organization such as the structure of the internal networks or software systems like operating systems, tools/utilities, or software applications used by the organization and then initiate connections to the internal network and carry out attacks. Intrusions are normally carried out by people outside the organization. Sometimes, intrusions can be caused by internal authorized persons carrying out these attacks by misusing their authorization or by internal authorized persons who go beyond their area of authorization and such attacks also need to be protected against.

An Intrusion Detection System (IDS) is a hardware/software combination or a combination of both hardware and software that detects the intrusions into a system or network. IDS complements a firewall by providing a thorough inspection of both the packets' header and its contents thus protecting against attacks, which are otherwise perceived by a firewall as seemingly benign network traffic.

Firewalls look at the control rules; a packet is either allowed or denied. A rule specifies whether a host or a network, or an application should be allowed into the trusted network. To check the rules, a firewall has to just inspect the header of the TCP/IP protocol such as FTP, HTTP, or Telnet. However, it does not inspect the data contents of the network packet. Even if the data contains a malicious code, the firewall will allow this packet to pass through as the packet header has conformed to the rules configured in the firewall. Hence, you can still have a firewall but your trusted network can be compromised.

IDS inspect each and every packet's content traversing the network to detect any malicious activity. Every packet is peeled all the way down to the "data content" part and the data content is inspected for any malicious code and then the packet is reassembled back to its original form and then the packet is sent along. As you can see, every packet is dissected and then assembled back at layer 3, which makes the IDS very process intensive when compared to the firewall.

The firewall is a necessary component of an overall network security topology but is insufficient on its own. Most of the modern networks have IDS as an essential part of the security architecture.

## Why Use IDS?

Why use IDS? The answer is very simple: We do not want somebody to enter our system with malicious intentions and carry out attacks on our systems, thereby endangering our whole network. We want to be alerted of any such activities so that we can act upon such incidents immediately and take actions to eliminate the root causes of such issues and eliminate any potential damages/disruptions caused by them.

Even if we have a good logging system of all the traffic entering into our internal network, it is tedious to go through these logs. It is manually impossible to differentiate between a malicious packet and a good network packet. Even with the help of computers, this is an intensive job which requires lots of processing power. We have seen that over the years, in this mostly connected world, connected through various means including tablets and mobile phones, bad people with bad intentions target various corporations as well as individuals. As it is impossible to detect such attacks manually to prevent or mitigate them, it has become imperative to have an automated tool to help us monitor the system for attacks. IDS has become a useful tool to provide this monitoring.

Before we begin our discussion, let's familiarize ourselves with some of the terms that are used in IDS/IPS technology:

- True Positives: These are alerts that something is not right when it is actually not right. Example: The IDS finds a packet as containing malicious code and it was actually true that the packet had malicious code, as confirmed by investigation.
- True Negatives: These are alerts that something is right when it is actually right. Example: The IDS finds a packet as containing no issues and it actually had no issues.
- False Positives: These are alerts indicating that something is not right with a packet when actually it is right. Example: The IDS finds a packet as having malicious code but it is actually a genuine code.
- False Negatives: These are alerts that something is right when actually it is wrong. Example: The IDS finds that a packet does not have any malicious code but it actually does contain a malicious code, as found through investigation.

Figure 11-1 provides explanations.

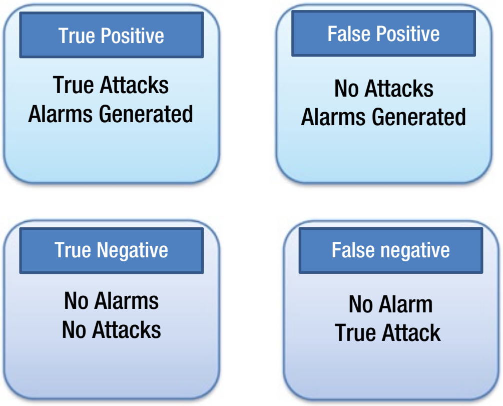

***Figure 11-1.*** *Definitions of IDS/IPS Alerts*

# Types of IDS

There are two types of IDS:

- Host-based IDS: Protects the end system or the network resources.
- Network-based IDS: Monitors network traffic for attacks. A Network IDS is deployed on the network near a firewall, on the DMZ or even inside the trusted internal network.

## Host-Based IDS (HIDS)

Host-based Intrusion Detection System refers to the detection of intrusion on a single system. This is normally a software-based deployment where an agent, as shown in Figure 11-2, is installed on the local host that monitors and reports the application activity. HIDS monitors the access to the system and its application and sends alerts for any unusual activities. It constantly monitors event logs, system logs, application logs, user policy enforcement, rootkit detection, file integrity, and other intrusions to the system. It constantly monitors these logs and creates a baseline. If any new log entries appear, HIDS checks the data against the baseline and if any entries are found outside of this baseline, HIDS triggers an alert. If any unauthorized activity is detected, HIDS can alert the user or block the activity or perform any other decision based on the policy that is configured on the system.

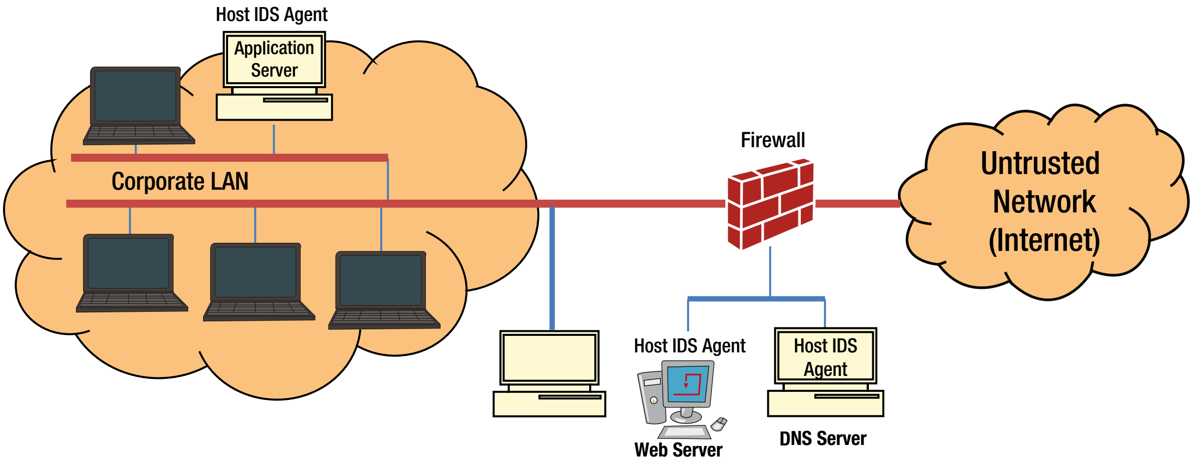

***Figure 11-2.*** *Host-Based Intrusion Detection System*

Most of the HIDS products have the ability to prevent attacks also. However, it is initially deployed in the monitor mode and then once there is an understanding of the system activity, a baseline is created and then HIDS is deployed in prevention mode. The functionality of HIDS depends on the logs generated by the system and the fact that the intruders leave evidence of their activities. Generally, hackers get access to the system and install malicious tools so that future access becomes easier. If these tools change the operating system configurations, or entries of some windows registry, it is logged in the systems/event log, thus triggering an alert by the HIDS system.[1]

HIDS is generally installed on servers, or end point devices to protect the system from intrusion. The function of HIDS solely depends on the audit trails generated by the system. If hackers manage to turn off these logs, even if you have a HIDS agent running, it may not trigger any alerts. This is the biggest disadvantage of HIDS.

Advantages of HIDS are:

- System level protection. Protects from attacks directed to the system
- Any unauthorized activity on the system (configuration changes, file changes, registry changes, etc.) are detected and an alert is generated for further action

There are disadvantages also:

- HIDS functionality works only if the systems generate logs and match against the pre-defined policies. If for some reason, systems do not generate logs, HIDS may not function properly
- If hackers bring down the HIDS server, then HIDS is of no use. This is true for any vulnerability protection software

## Network-Based Intrusion Detection System (IDS) / Intrusion Prevention System (IPS)

A Network-Based Intrusion Detection System (NIDS) [1] monitors (and detects) any suspicious activity on a network. It checks each and every packet that is entering the network to make sure it does not contain any malicious content which would harm the network or the end system. Network Intrusion Detection System sniffs the network traffic continuously. The traffic is matched against known signature profiles and if there are any abnormalities found in the traffic, then a NIDS triggers an alarm to the management console. A single sensor, as shown in Figure 11-3, deployed in promiscuous mode or inline mode can monitor/protect several hosts in the network.

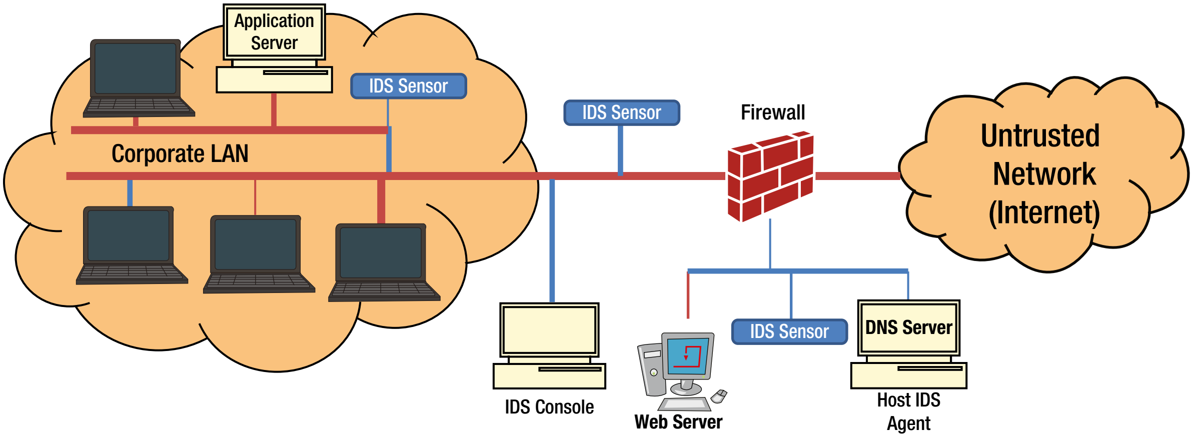

*Figure 11-3. Network-Based Intrusion Detection and Prevention System*

Network IDS protects the network and its resources from the network perspective. For example, network IDS can detect reconnaissance attacks, Denial of Service attacks right at the network level. NIDS generates alerts as soon as it discovers these attacks. NIDS is a hardware/software solution placed near the firewall as an independent device (sensor) and has network operating system (TCP/IP stack). Sensors have interfaces to monitor the network (monitoring interface) and a management interface which is used for controlling and receiving alerts and for sending these alerts to the central management controller.

An **Intrusion Prevention System (IPS)** is used to prevent the intrusion. It is an extension of IDS. IDS only detects whereas IPS protects the network from intrusion by dropping the packet, denying entry to the packet or blocking the connection. IPS and IDS together monitor the network traffic for malicious activities and IPS is considered as just an extension of IDS. The main difference is that the IPS are placed in-line to prevent intrusions and IPS can take decisions like dropping the packet, or resetting the connection along with sending alarms to the management console. An IPS can also detect/correct fragmented packets, Cyclic Redundancy Check (CRC) errors, or TCP sequencing issues.

Table 11-1 summarizes the key differences between the IDS and IPS. Today, most of the network-based intrusion systems combine both detection and prevention – Intrusion Detection and Prevention Systems (IDPS).

***Table 11-1.*** *Key differences between IDS and IPS*

| Intrusion Detection System (IDS) | Intrusion Prevention System (IPS) |
|---|---|
| Passively monitors network behavior and "detects" attacks | Actively analyzes network behavior and "prevents" attacks in real-time |
| Supports both Network and Host level detection | Supports both network and host level detection |
| Passive monitoring, does not sit in the data path | Active monitoring, deployed in-line mode |
| Key measure is detection accuracy | Key measure is lesser number of false positives |
| NIDS: ISS, Cisco, Enterasys, Symantec | NIPS: McAfee Intrushield, NetScreen, Tippingpoint. |
| HIDS: ISS, Symantec, Enterasys | HIPS: Cisco, McAfee (Entercept). Snort – an open source Network IDS/IPS developed by Sourcefire |

Table 11-2 summarizes the pros and cons of Host-based IDS and Network-based IDS.

***Table 11-2.*** *Pros and Cons of H-IDS and N-IDS*

| | Pros | Cons |
|---|---|---|
| Host IDS | Protects from attacks at the host level<br>No Bandwidth Impact | Impacts host resources – CPU, memory<br>Operating System dependent<br>One agent can protect one host only |
| Network IDS | Protects network and network resources<br>Protects against DoS attacks | Sensor hardware is process intensive<br>Prone to false positives. |

# How Does Detection Work?

The Intrusion Detection and Prevention Systems detect intrusions through the following mechanisms: signature-based detection, anomaly-based detection, or stateful protocol analysis. In this section we will look at each of the methods in detail.

## Signature-Based Detection

This mechanism protects against known threats. A signature is a known pattern of a threat, such as:

- An e-mail with an attachment containing a known malware with an interesting subject (for example, an e-mail with the subject "I love you").
- A "remote login" by an admin user, which is a clear violation of an organization's policy.

Signature-based detection is the simplest form of detection because it just compares the traffic with the signature database. If a match is found then the alert is generated, if a match is not found then the traffic flows without any problem.

In signature-based detection, detection is based on comparing the traffic with the known signatures for possible attacks (see Figure 11-4). They can only detect known threats and hence, are not efficient in detecting unknown threats. To detect an attack, the signature matching has to be precise, otherwise, even if the attack has a small variation from the known threat signature, then the system will not be able to detect. For example, in the above example, instead of "I love you" if the subject is "love you", the system may not detect the threat. Hence, it is very easy for the attackers to compromise and breach into the trusted network.[2]

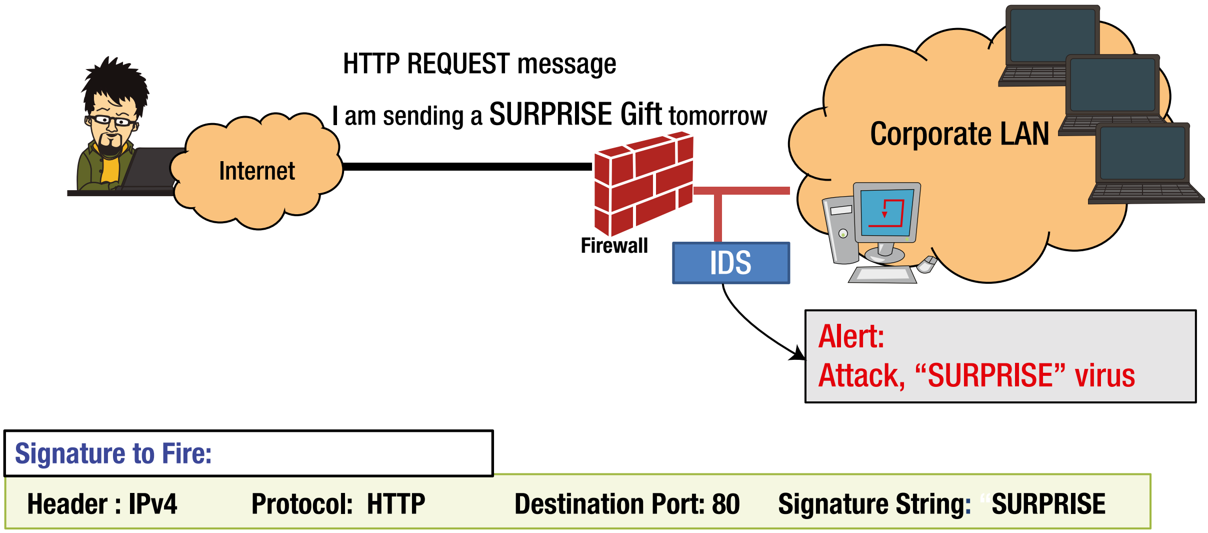

***Figure 11-4.*** *Signature-Based Detection (Flow diagram)*

Signature database needs to be updated constantly, almost on a daily basis from the anti-virus labs such as McAfee, Symantec, TrendMicro, and other security providers. If the signature is not up to date, chances are that the IDS systems will fail to detect some of the intrusion attacks. The other disadvantage is that they have very little information about previous requests when processing the current ones.

Signature-based detection can offer very specific detection of known threats by comparing network traffic with the threat signature database. The detection can be enhanced if the network traffic inside the network can be made to learn specific patterns, thus reducing false positives. Signature detection engines tend to degrade in performance over a period of time as more and more signatures are added to the database. It takes more and more time for the engine to do a pattern search as the signature database is always growing as more and more definitions are added to it. Hence, a robust platform is needed for signature detection considering this growth. Table 11-3 summarizes the pros and cons of signature-based detection technique.

***Table 11-3.*** *Pros and Cons of Signature-based Detection Technique*

| Pros | Cons |
|---|---|
| Simple method to create | High false positives rate |
| Applicable across all protocols | High false negative rate |
| | Multiple signatures are required for a single vulnerability |

## Anomaly-Based Detection

Anomaly-based detection (see Figure 11-5) protects against unknown threats. An “anomaly” is anything that is abnormal. If any traffic is found to be abnormal from the baseline, then an alert is triggered by the IDS suspected of an intrusion. IDPS first creates a baseline profile that represents the normal behavior of the traffic. The baseline profile is created by allowing the IDS system to learn the traffic over a period of time so that IDPS can study the traffic behavior during peak hours, non-peak hours, night hours, early hours of business, and as per your organizational network behavior. After learning, the traffic collected over a period of time is statistically studied and a baseline profile is created. Once the IDS is changed from learning mode to detection/prevention mode, it starts comparing the regular traffic with the profile that was created, and if any abnormality or deviation from the baseline profile is found, then an alert is triggered cautioning the possible intrusion or the intrusion is prevented, if it is configured for prevention mode. Customized profiles can also be created for specific traffic behavior such as the number of e-mails sent by a user and user access attempts.

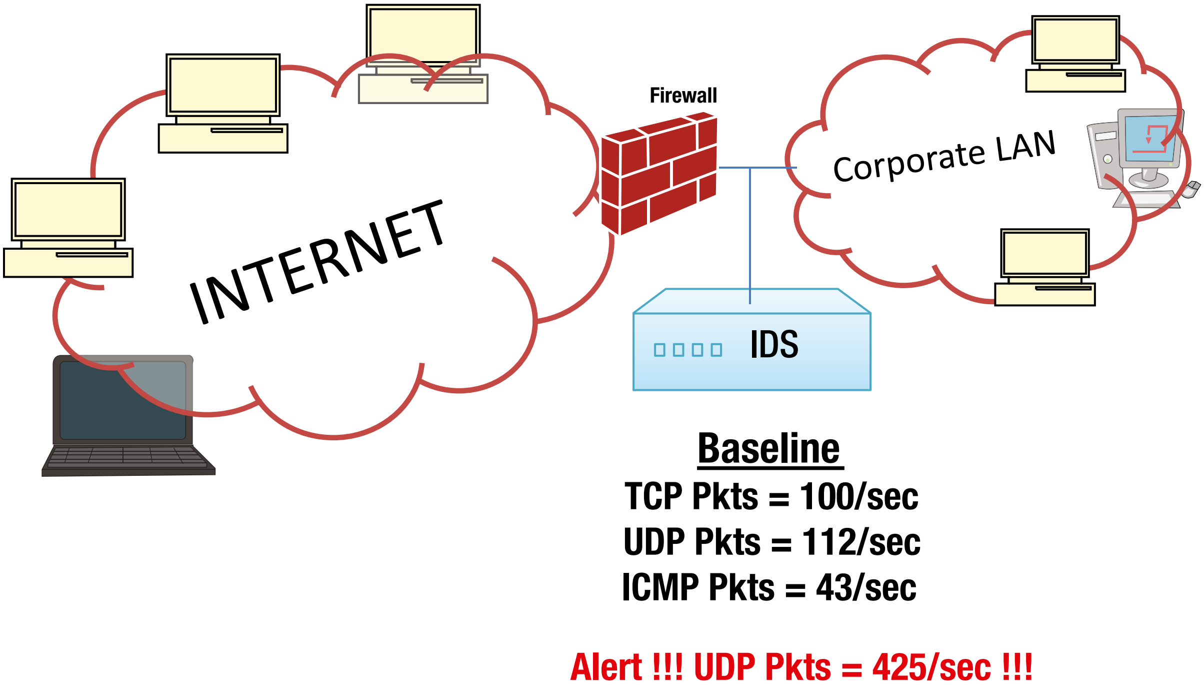

***Figure 11-5.*** *Anomaly Detection*

What is an anomaly? Here are some examples of anomalous behavior:

- Too many Telnet sessions on a single day
- HTTP traffic on a non-standard port
- Heavy SNMP traffic

For effective intrusion detection, IDS must have a robust baseline profile which covers the entire organization's network and its segments. It should cover normal traffic behavior of all the components which are aimed to be covered by the Intrusion Detection and Prevention System. Baseline profile can vary in complexity from a simple to a comprehensive content, depending on the characteristics of the network and its components. For example, a profile could include the following data:

- A web application logged in remotely by a specific set of users
- An application which has a specific acceptable password design
- Traffic during the peak hours and non-peak hours as defined by the organization
- Connectivity pattern from an external partner network
- Connecting from a set of mobile devices to the database server

The challenge of the anomaly-based detection method is creating an effective profile. The initial profile, sometimes referred to as the "training profile," is generated by studying the traffic pattern over a period of time. The time factor may vary from organization to organization. It could be a few hours to a few days. Once this profile is created, IDS is put into detection mode and every time there is a packet, a pattern is matched against the baseline profile. This baseline can be changed as and when required based on the traffic behavior. If any malicious activity already exists from the beginning, while building the baseline profile, this activity will also become part of the baseline profile and such kind of malicious activity will thereby go undetected. Hence, anomaly detection does not necessarily detect each and every unknown attack. The limitation is based on the baseline profile you create. However, a system administrator was alerted by IDS to Microsoft DCOM DOS vulnerability without having a specific signature.

## Types of Anomaly

Anomaly-based Intrusion Detection and Prevention Systems (IDPS) protect anomaly caused due to violation of protocols, and application payload. It also protects against Denial of Service attacks and Buffer overflow attacks.

### Protocol Anomaly

Protocol anomaly refers to the anomaly in the protocol format and protocol behavior with respect to the Internet standards and specifications. There are many aspects in TCP and IP protocol that needs to be monitored, for example, different flags, SYN, ACK, and FIN, and their combination in TCP header and the reserved flags of IP header. The way IP fragmentation and reassembly is implemented is as per the standards. If this anomaly is not detected by the IDS, the end host may not process these unconventional packets and this may lead to the crash of the system. At the application level, IDPS must be able to do deep protocol parsing to understand application level protocol anomaly. It also requires a deep understanding of the application semantics in order to detect application payload anomaly.

Some other examples include:

- Unusual TCP segmentation and TCP flags combination
- Corrupt checksum
- Incorrect IP fragmentation and reassembly flags
- Erroneous source and destination port numbers
- Illegal protocol commands and its usage
- Running protocol on non-standard port
- Presence of shellcode in unexpected application protocol fields
- Misuse of protocol and protocol services

### Statistical Anomaly Detection – Statistical DDoS

Denial of Service (DoS) and Distributed Denial of Service (DDoS) results in a burst of traffic on the network which is not normal. In order to overcome this kind of attack, baseline profiles are created on the normal flow of traffic, as described earlier, based on statistical modeling, such as Naïve Bayes, to determine anomalous packets on the network. While learning the network traffic behavior, the function of statistical modeling is to compute the probability score for each of the data packets that is considered as normal traffic. The scores are computed based on the sampled data over a period of time and stored in a baseline profile. A threshold is set for each set of protocols and users. When the IDS is in monitoring mode, the data is checked against the baseline and the threshold. Whenever an anomalous packet is discovered and the scores are above threshold, then an alert is triggered. The reporting process will report only when the data is found to be anomalous for a sufficient period of time; otherwise, the IDPS will simply ignore the trace. Threshold can be set for different profiles, for different protocols, and for different users.[3,4]

When IDPS is in monitoring mode, if there is anything that is abnormal to the baseline, the system will generate an alert. But, it may turn out that the analysis results confirm that the alert found was a false positive. As a security administrator, one can expect a similar kind of traffic behavior appearing every other day and to minimize the spending of the same effort repeatedly, a threshold can be set so that anything within this threshold, the traffic is still considered normal and anything which exceeds this threshold is considered an intrusion. Thresholds can also be set for a set of users, or set of protocols.

Profiles based on the statistical measures can detect some of the DoS anomalies based on long- and short-term distributions or bursts of peak (i.e., high) traffic. The normal baseline profiles are continuously being learned while the system is in detection mode and the baseline is re-created to adjust the changing traffic pattern to avoid false positives.

By creating different profiles, DoS attacks can be prevented. For example, for each of the DoS attacks, a profile can be created. Knowing the pattern of SYN flood, a SYN flood DoS profile can be created. Whenever there is SYN flood traffic on the network, the IDS sensors can detect the SYN flood attack by comparing the network traffic with the SYN flood profile thus alerting a SYN flood attack. Similarly, UDP flood profile, TCP data segment profile, or ICMP flood profiles can be detected and alerted.

Though anomaly-based IDS has an advantage of detecting unknown attacks, defining rules for it is difficult. Each protocol must be analyzed, processed, and compared with a baseline. Any customized protocol makes it even more challenging.

Another major pitfall of anomaly detection is defining normal traffic while creating a baseline. Normal traffic has to be clean and should not have any malicious activity in the network. In case of any malicious activity during the learning process, then the baseline profile learns this and makes it harder to detect this intrusion or it may not even detect intrusion of such malicious traffic. For example, reconnaissance attacks such as fingerprint or directory traversal, which complies with network protocol, easily goes unnoticed since it complies with protocol and payload limitations. Some of the pros and cons of statistical anomaly detection are summarized in Table 11-4.

***Table 11-4.*** *Pros and Cons of Statistical Anamoly Detection*

| Pros | Cons |
|---|---|
| Detects Unknown Attacks | Prone to false positives |
| Prevents DoS attacks, Buffer Overflows | Longer detection time |
| | Analyzing Intrusion may be difficult with Anomaly |
| | Difficulty in creating baseline |

## Stateful Protocol Analysis Detection

This method is similar to the anomaly-based detection, except that the profiles are created by the vendors who supply the sensor equipment (IDPS). The profiles are predetermined and made up of the generally accepted benign network traffic activity as specified by the standards. “Stateful” means that the IDPS has the capability to keep track of the state of the protocol both in network layer and application layers. For example, in case of a TCP connection establishment state, the IDS should remember all the connection states. Similarly, in case of authentication, the initial connection session is in an unauthorized state and IDS should remember these states. After an exchange of some information between the two parties, the client and the server, the user is authenticated and allowed access to the network. During this period, the traffic is benign and the IDPS should remember the state or it will lead to false positives.

The stateful protocol anomaly detection method uses profiles that have been created based on standards and specifications specified by the vendor who generally complies with most of the protocols from the standard bodies (Internet Engineering Task Force). If any vendor has implemented protocols, with variation to the standards, it would cause difficulty for the IDPS in detecting and analyzing the states. In such cases, IDPS protocol models also need to be updated for the customized protocol changes.

The primary drawback of this method is that they are process- and memory-intensive like many protocols, and the IDPS has to keep track of their states simultaneously. Another problem is if an attack is within the generally acceptable protocol behavior, then it can pass through. If the protocol implementation varies from operating system to operating system then IDPS may not perform well in detecting the intrusions. The pros and cons of this method are summarized in Table 11-5.

***Table 11-5.*** *Stateful Protocol Detection*

| Pros | Cons |
|---|---|
| Stateful Inspection | Resource intensive |
| Reasonable checks on the standard protocol before an alert | Cannot detect variations to the generally acceptable protocol behavior policy |
| | Cannot detect any conflict between the standards and how they are implemented |

## IDS/IPS System Architecture and Framework

The architecture of a typical network-based IDS/IPS is as shown in Figure 11-6. It consists of a hardware device, management console, a database, and connectivity to network management consoles.

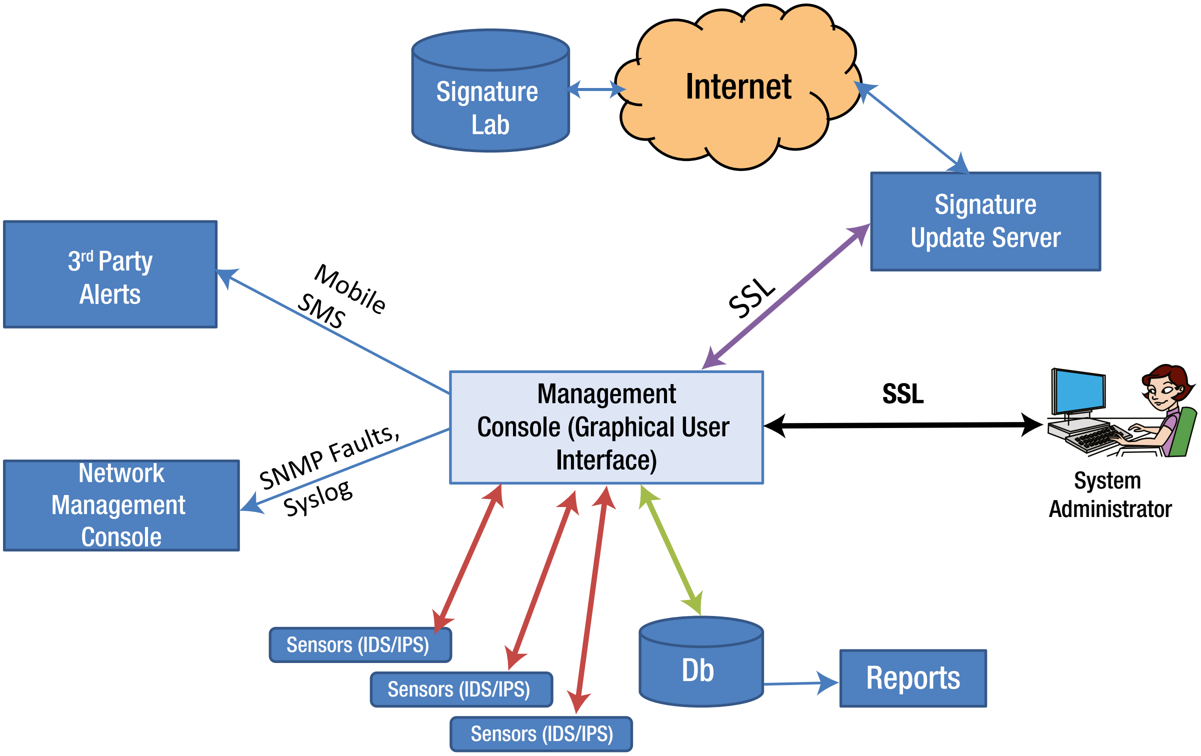

***Figure 11-6.*** *IDS/IPS Architecture*

## Appliance (Sensors)

The primary function of a sensor is to analyze traffic and respond when the attacks are detected. The sensor examines each and every packet's header and data content that enters the network. The sensor looks for a pattern and behavior in the network traffic that indicates malicious activity and sends alerts to the management console. The sensor examines the packets and checks against the user-defined policies or rule sets, which contains the priorities of the attacks to be monitored and the counter measures to be taken when an attack is detected.

If an attack is detected, the sensor sends an alert to the management console, logs the alert, and responds to the attack as per the defined policy. The policies for sensors can be configured to several types of responses - generating alerts, logging events, resetting TCP connections, blocking traffic at firewalls, scrubbing malicious packets, and even dropping the packets entirely before reaching the final destination.

DatabaseA database server is an important component of the overall IDS/IPS architecture. It is a repository of all the events triggered by the sensors, logs generated, user policies and profiles, and other functional information.

Management ConsoleA management console provides an interface to the users and administrators for configuring and managing sensor systems. The users connect to the management console through a client system over a web interface or any other client software. A Management Graphical Interface should provide the following:

- **Alert/Event Viewer:** Displays all the intrusions detected by the sensors, which have violated the defined set of policies. The alert viewers should be able to provide drill-down capabilities to view all the details of individual alerts such as host, destination, service, type of attack, and action taken.
- **Incident Generator:** This enables the creation of real-time correlative analysis of attacks on the network. This should provide the type of incident that has occurred and when it has occurred.
- **Report Generators:** To generate various security reports for the management and further analysis. It should have the capability of generating reports automatically as well e-mailing them to individuals.
- **System Configuration Tools:** Provides all the system configuration features. Setting polices, profiles, responses to attacks, sensor mode of operation, user created profiles, baseline scheduling, defining user roles and responsibilities, sending alerts to central network management console, and other sensor level configurations. It should also have the capabilities to send alerts to the central network management console and alerting administrators through triggering cell phone calls and SMS services.

## Signature Update Server

For the IDS system to detect the latest vulnerabilities and threats, its threat signatures must be up to date. Sensors should be updated with the latest signatures regularly. Both the management console and the sensors should always be up to date with the threat signature set and software patches. A good system should have the capability to connect to the latest threat vulnerability lab(s) and download the signatures on a regular basis and update the sensor(s) that are detecting the intrusions.

In the architecture illustrated, the signature update server connects to the signature library to download the latest signature set and then the management console pushes the updated signature to the sensors. The polling interval of the update server, and the signature push to the sensors can be configured using the management user interface. Once the update server receives the new signature updates, the management interface determines what signatures need to be pushed onto the sensors based on the policies that are defined and applied. For example, a policy defined for a particular LAN segment will only update those signatures defined for that LAN segment (for example Windows security patches).

It is always advisable to configure the update server to get the latest signatures as soon as they are available to improve the overall level of protection, and having an automatic access to the signature lab reduces operational overheads.

Some of the other capabilities the architecture should support include[1]:

- **Logging capabilities:** Should support logging related to intrusion detection, incidents, and other system-related information and should be able to categorize the severity, the impact, and the priority of the intrusions and provide information regarding the prevention actions it has taken. The system architecture should have capabilities to store logs both locally and at a central repository and should have the capability to synchronize time with Network Time Protocol.
- **Detection Capabilities:** Should have broad and extensive detection capabilities; up-to-date threat signatures; the flexibility of customization and fine tuning of the baseline profiles and user-defined profiles to improve detection capabilities; the capabilities to set threshold limits to minimize false positives; and be able to block a connection after a set of failed connection time (retries).
- **Code viewing and editing capabilities:** Technologies should support viewing the virus code or threat code to understand the nature of the threat. This helps in writing a customized signature locally.
- **Prevention Capabilities:** It should have the flexibility to configure prevention capability for each type of attacks. It should support recommendation for prevention for certain unknown attacks and DoS attacks. This helps the administrator to fine tune the policy and reconfigure the sensors.

## Attack types Detected by IDS

An Intrusion Detection and Prevention System (IDS/IPS) is a software/hardware combination that detects intrusions and if appropriately configured, also prevents the intrusion. An IDS inspects each and every packet entering the network by peeling off the packet header and its contents and doing a thorough inspection of the packet before allowing the packet into the network. It complements a firewall and the anti-virus software and protects against any attacks embedded within the packet data which goes unnoticed by a firewall.

The Intrusion and Detection System (IDS) should detect all the types of attacks, including Reconnaissance, Denial of Service (DoS)/Distributed Denial of Service (DDoS) and other network attacks, using techniques such as signature-based detection and anomaly-based detection. It should detect both known and unknown attacks. Table 11-6 summarizes the type of attacks that an IDS/IPS detects/prevents.

***Table 11-6.*** *Attacks detected by IDS/IPS*

| Attacks Detection by IDS/IPS | Attack Type |
|---|---|
| Shellcode in password<br>Too many strange IP fragments<br>Too much UDP than TCP<br>Many HTTP requests than responses<br>Buffer Overflow | Anomaly (Unknown attacks) |
| TCP SYN Flood attack<br>ICMP Flood<br>TCP or UDP Flood<br>Ping of Death<br>Smurf attacks<br>Winuke<br>Apache2<br>Back<br>Teardrop<br>SYN Flood<br>UDPStorm<br>IP Spoofing | Denial Of Service (DOS)<br>and<br>Distributed Denial Of Service (DDoS) |
| IP fragmentation overlap, options, etc.<br>TCP segmentation overlap, options usage<br>All checksum/length consistency | Protocol Anomaly<br>Transport layer reconnaissance and attacks |
| DNS request – Illegal field value and combinations<br>HTTP, SNMP, SMTP – Illegal use of commands<br>Unusually short or long field lengths<br>Unknown protocol port numbers – Gnutella on port 80, HTTP on port 89<br>URL encoding<br>SQL Injection Attack<br>Buffer Overflow<br>Telnet/FTP escape sequence attacks | Application protocol anomaly |
| Nmap<br>MScan<br>Satan<br>IPSweep<br>Fingerprint<br>Port scan/Network Scan | Reconnaissance |

## Responses by IDPS to the Intrusions

Intrusion Detection System (IDS) only detects intrusion. It sends alerts to the management console. It detects both known and unknown attacks by inspecting each and every packet that enters the network. However, IDS does not take any action by itself to protect the system or network. Whereas, Intrusion Prevention System (IPS), not only detects the intrusions, it also prevents the intrusions by taking one of the following proactive steps:

- **Block or Deny the packet:** When the next packet arrives from the same source, IDPS can simply block that particular user's data packets entering the network by automatically configuring the sensor to "block." The intended bad packet never reaches the destination and it is blocked at the entrance itself.
- **Reset connection:** Reset the session when the next packet arrives from the same source. Close the session of the intrusion source. The goal is to terminate the attack before it succeeds. When the attack is detected, RESET connection instructions should be sent to the host in the trusted network. Unfortunately, if the RESET packets are not received in time by the host in the trusted network, then the attacker may succeed. RESET is applicable only for TCP packets and cannot be used for UDP or ICMP packets.
- **Dropping the packet:** Completely drop the packet with intrusions. As soon as the intrusion is detected, identify the source and automatically configure the sensor to drop the packet from that source. The bad packets never reach the intended destination.
- **Reconfigure firewall:** Depending on the type of deployment and where the sensor is deployed, as soon as the intrusion is detected, IDPS can instruct the firewall next to it to change the "Access rules/policies" to deny the packet from the intrusion source, thus preventing any attackers from succeeding.

## Deploying IDS/IPS

IDS/IPS is typically deployed either in detection mode or both detection and prevention mode. In detection mode, it is deployed to monitor the network. It sits in the network just like any other component and reads the data in silent mode without affecting the regular traffic. Whereas, the IDS/IPS prevention mode is deployed in such a way that the normal traffic passes via IDS/IPS. The detection mode is called the **Passive Mode** and the second mode is called the active mode or in-line mode.

Where and how you deploy the sensors depends on the organization security policy and its network topology. Requirement depends on the type of assets, configuration of the network, location of the aggregation points, type of traffic, and so on. Initially, it is always recommended to deploy in the detection mode. Once the IDS is able to understand the network and the baseline profile is stabilized, then it is better to move to the prevention mode.

### Passive Mode

In passive mode, the traffic does not pass through the sensors. In this mode, the sensors monitor a copy of the actual network traffic. Each and every packet will be read by the sensor for any intrusion. In case of an intrusion, an alert is triggered and prevention measures are taken by the security administrators. In passive mode, the sensors are deployed in what is called as the **SPAN mode**.

## Span Mode

Each switch comes with a SPAN port that can monitor all the network traffic going through the switch. A sensor is connected to this SPAN port, as shown in Figure 11-7, so that all the traffic flowing through the network, including the host's, the server's, and the client's can be monitored. If the switch is not configured properly or reconfigured, then the traffic may not be adequately monitored by the sensors. In case of heavy traffic, if the SPAN port is disabled by the switch, this again becomes a problem as the sensor will not be able to monitor the intrusions properly.

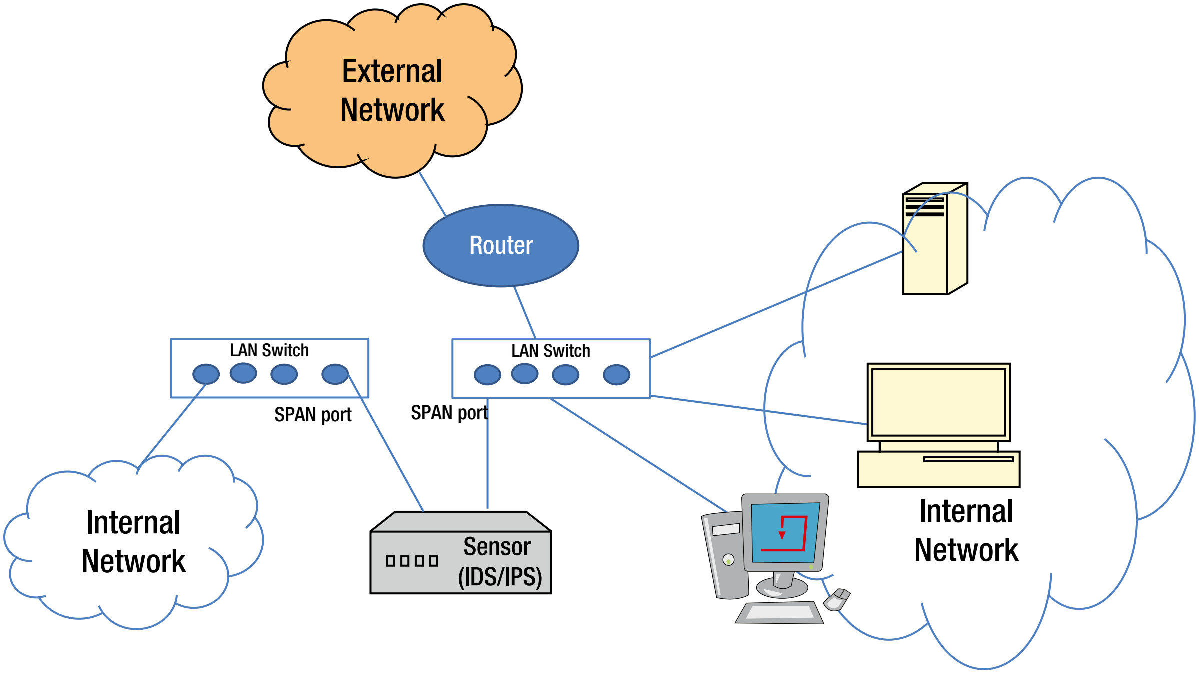

***Figure 11-7.*** *SPAN mode*

## Tap Mode

In this mode, network taps are placed on a single wire for a particular segment of the entire network. The sensors monitor the traffic on the tapped wire by copying every network packet that comes through the tap. Whenever there is an intrusion, alarms are triggered. Preventive measures are taken by the security administrators. Network taps are a special device connected to the media as shown in the Figure 11-8.

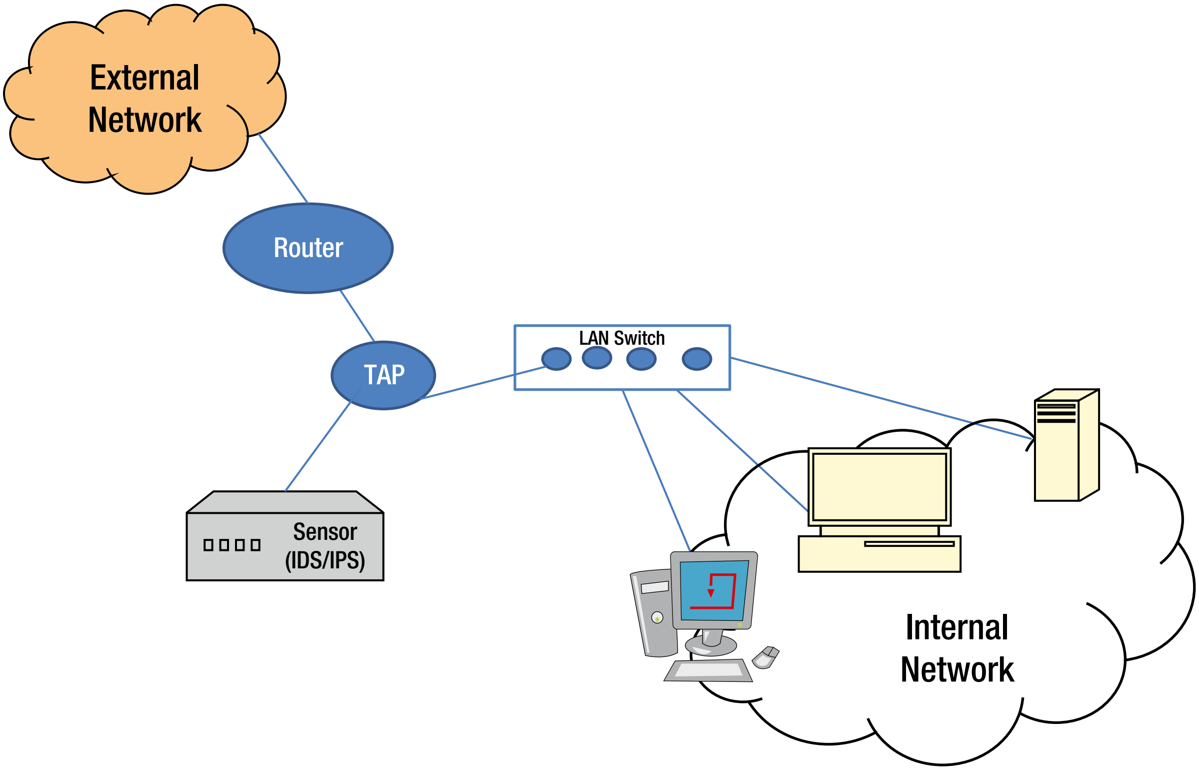

***Figure 11-8.*** *Network Tap mode*

## In-Line Mode

Sensors are placed directly in-line with network traffic as shown in Figure 11-9. Network traffic will pass through the sensor in real-time thus preventing intrusions into the trusted network. In-line sensors are placed next to the firewall and other network/security devices. It is placed with a clear distinction between internal network and external network. It is often deployed near the DMZ area where it is more secured from the internal network and has less traffic to process. Sensors can also be deployed inside the network to monitor the traffic that is going around inside the network and the traffic that is going out of the organization’s systems. The deployment architecture solely depends on the organizational need and security architecture.

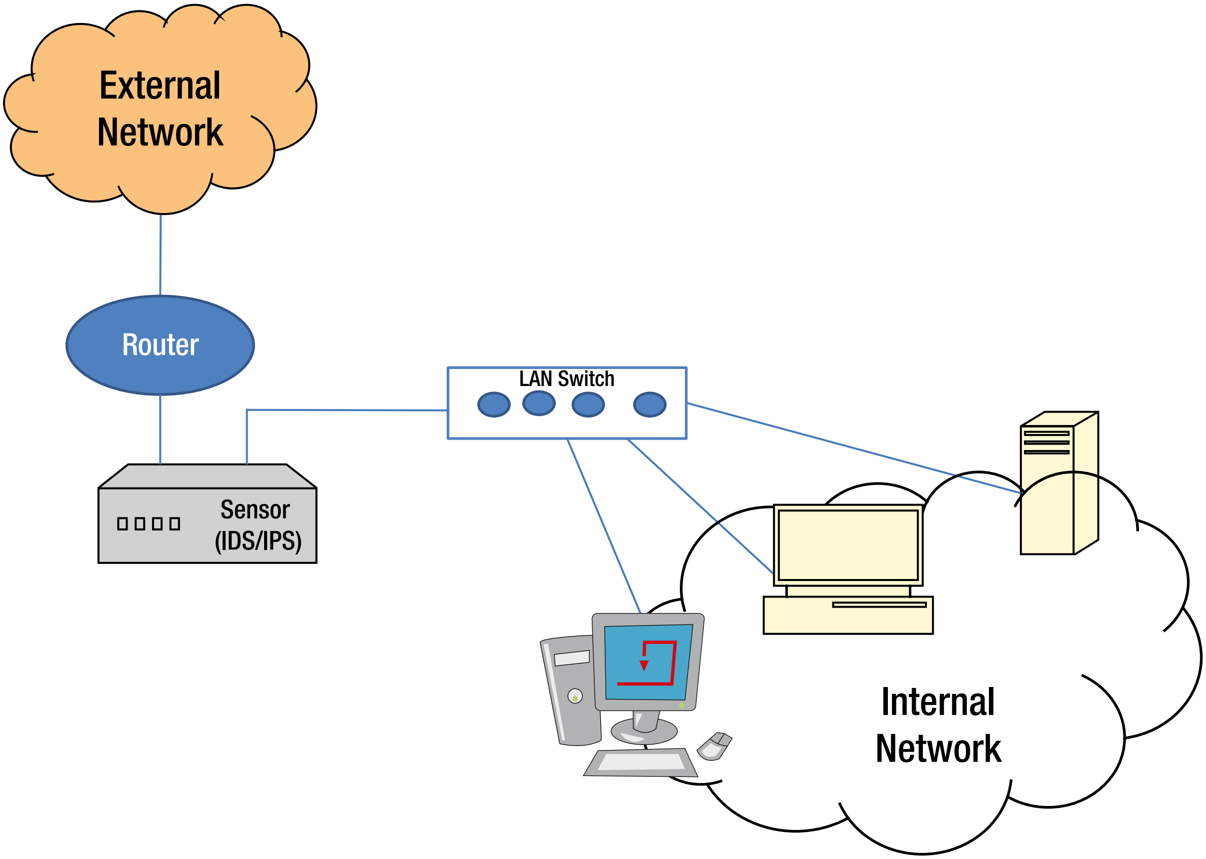

***Figure 11-9.*** *In-line mode*

The benefits of using sensors in in-line mode are as follows:

- **Intrusion Prevention:** During in-line mode, the sensors are in prevention mode either by blocking the traffic or dropping the packets in case of intrusion, thus preventing malicious packets reaching their intended destination. The sensors can be configured for countermeasures such as reset connection, reconfigure firewall, or block the traffic as soon as they detect any intrusions by mediating the traffic flow.

  However, the risk of in-line mode is the granularity of identifying the malicious packets. The sensors should be designed to take preventive measures only against those packets that are malicious. Reconfiguring a firewall with false positives can prevent genuine traffic from entering or leaving the trusted network.

- **Processing at wire speed:** Sensors deployed in in-line mode should process packets at wire-speed, otherwise the traffic passing through the sensors can become a bottleneck and hinder the network performance.

- **Traffic Normalization (Packet Scrubbing):** Though the baseline profiles have been created with what is perceived to be normal traffic, sometimes ambiguities such as IP fragmentation can cause false positives. The sensors can reassemble IP fragments, TCP segments, at the sensor level, normalize the traffic, re-evaluate the profiles, and improve upon false positives.

  It is important to deploy in-line mode sensors in a high-availability state. In in-line mode, there are high chances of the sensors becoming single points of failure which will result in a complete breakdown of the network. If a network is running in in-line mode, it is recommended to have two sensors in a high-availability mode as shown in the Figure 11-10.

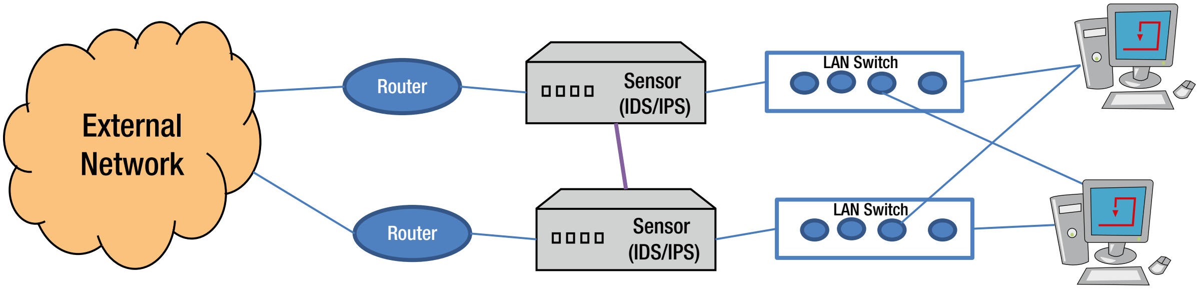

***Figure 11-10.*** *In-line mode - High Availability*

## IDS/IPS in Context

IDS/IPS complements both the firewall and the anti-virus software. IDSs provide protection to known attacks, unknown attacks, and DoS attacks. IDS peels off each network packet that enters the network, and inspects the data for any abnormalities, which a firewall is unable to do. However, a firewall acts as a basic security guard, who checks your identity and legitimacy before allowing you to enter the network, whether you are internal or external to the organization. Intrusion Detection and Prevention System (IDPS) is ideally deployed in monitoring/detection mode. If an IDPS lacks proper rules and the threat signatures are not up to date, then IDPS neither detects the intrusions nor prevents them. It is extremely important to have the threat signatures updated on a regular basis and connected to the signature labs directly.

Strategic placement of IDS/IPS sensors in the network is also crucial. Whether it is in tap mode, span mode, or in-line mode is dependent on the organizational security and network policy. In order to deploy in in-line mode, one should have more experience in configuring IDPS to have different prevention modes thereby avoiding too many false positives and blocking genuine packets. Also, for in-line mode, the sensors should support wire-speed processing capabilities. For example, if your network is 10 Gbps, the sensors should process at the same speed. Another important deployment strategy is location of IDPS – whether in DMZ, or before the DMZ, or immediately after the firewall or VPN (Virtual Private Network) concentrator. Since the traffic through VPN tunnels is encrypted, unless the IDS/IPS has keys, it may not be capable of conducting adequate analysis.

The most important is the remediation process. IDS can detect intrusions and send an alert to the security administrator but it has no control over the remediation. Similarly, IPS can take action to prevent intrusions but it has to be configured by the security administrator for each of the traffic and attack types. This is part of the process followed in an organization and cannot be automated. If the organization does not have means of responding to security incidents and alerts in a timely manner, having such systems may be useless.

In conclusion, firewall, anti-virus software, host-integrity system, and IDS/IPS technology complement each other. Having a firewall at the entrance of a network protects from unauthorized traffic entering and acts as a security guard; then IDS/IPS scrutinizes the traffic, acts as a surveillance system, and finally the anti-virus software keeps the system clean from further spreading of malicious software. It is crucial to have a layered defense in-depth strategy considering the threats being constantly created in the world of the Internet. The security is an overall process involving more than just technology. Technology can solve only a part of the problem but a process needs to be in place to protect the organization's assets from intrusion attacks.

## Chapter Summary

- We first defined intrusion in lay terms. Then we mentioned that IDS helps to detect intrusions and differentiated it from a firewall. We also learned that IDS peels off the packet and inspects it to understand whether the packet has any malicious code or can lead to any malicious activities. We also mentioned that IDS complements a firewall by doing what a firewall cannot do.
- We looked at why we need to use IDS. We mentioned that IDS not only provides alerts on intrusions but also enables us to take appropriate actions including corrective actions, based on root causes, to eliminate such intrusions in the future. We looked into a few of the important terminologies like false positives, true positives, false negatives, and true negatives in the context of the results of IDS.
- We then explored both the important types of IDS: host-based IDS and network-based IDS/IPS. We went through the details of host-based IDS including how it monitors the access to the system, its application, and sends alerts for any unusual activities. We then explained that it constantly monitors event logs, system logs, application logs, user policy enforcement, rootkit detection, file integrity, and other intrusions to the system. We then explained how the changes in logs can be interpreted by IDS and alerts are provided by IDS against any intrusions. We looked into the context of network-based IDS/IPS in that it inspects the network packets and checks against the stored malicious signatures to determine whether a packet has been sent with a malicious intention or not. We then differentiated between IDS and IPS. We further explored the pros and cons of both host-based IDS and network-based IDS/IPS.
- We explained how the signature-based detection and anomaly-based detection are used by IDS to identify the intrusions and provide the alerts. We then explored Protocol Anomaly and Statistical Anomaly Detection. We also looked into the advantages and disadvantages of the Anomaly-based Detection.
- We then explored Stateful Protocol Analysis Detection and listed the pros and cons of this form of detection.
- We explored on the architecture of IDS/IPS. In this context, we looked into the functions of important components of the IDS/IPS like Appliance (Sensors), Database, Management Console, and Signature Update Server, including the need to keep the signatures updated so that the detection is appropriately ensured. We also looked into the important capabilities of the IDS/IPS of needing to have logging capabilities, detection capabilities, prevention capabilities, code viewing, and editing capabilities.
- We then discussed various attacks the IDS/IPS can detect and prevent. We further discussed the various responses of IDS/IPS, including blocking, denying, or dropping a packet; resetting the connection; or reconfiguring the firewall.
- We discussed various modes in which the IDS/IPS can be deployed like SPAN mode, TAP mode, and in-line mode. We also looked at how IPS needs to be supported by wire speed processing in in-line mode.
- We ended the chapter with a final note on how firewall, IDS/IPS, and anti-virus play complementary roles to each other.