Chapter 7

Topologies and IDS

Solutions in this chapter:

- **Security Topologies**
- **Intrusion Detection**

- ☑ **Summary**
- ☑ **Solutions Fast Track**
- ☑ **Frequently Asked Questions**

Introduction

In today's network infrastructures, it is critical to know the fundamentals of basic security infrastructure. Before any computer is connected to the Internet, planning must occur to make sure the network is designed in a secure manner. Many of the attacks that hackers use are successful because of an insecure network design. That is why it is so important for a security professional to use the secure topologies and tools like intrusion detection and prevention that are discussed in this chapter. For example, if you are working with Cisco technologies (and other switch vendors), you might be familiar with virtual local area network (VLAN) technology. VLANs are responsible for securing a broadcast domain to a group of switch ports. This relates directly to secure topologies, because different Internet Protocol (IP) subnets can be put on different port groupings and separated, either by routing or by applying an access control list (ACL) (e.g., the Executive group can be isolated from the general user population on a network).

Other items related to topology that we examine in this chapter include demilitarized zones (DMZs). DMZ's can be used in conjunction with network address translation (NAT) and extranets to help build a more secure network. We'll look at each of these items and examine how they can be used to build a layered defense.

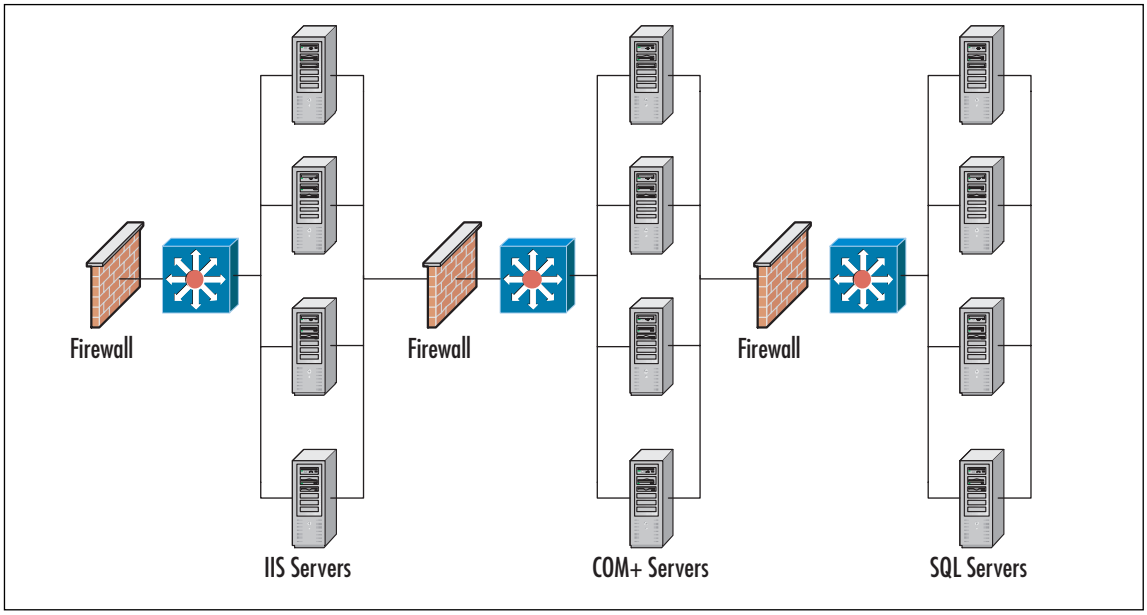
The second half of this chapter covers intrusion detection. It is important to understand not only the concepts of intrusion detection, but also the use and placement of intrusion detection systems (IDSes) within a network infrastructure. The placement of an IDS is critical to deployment success. This section also covers intrusion prevention systems (IPS), honeypots, and incident response.

Security Topologies

Not all networks are created the same; thus, not all networks should be physically laid out in the same fashion. The judicious usage of differing security topologies in a network can offer enhanced protection and performance. For example, suppose you have an e-commerce application that uses Internet Information Servers (IISes) running a custom Active Server Page (ASP) application, which calls on a second set of servers hosting custom COM+ components, which in turn interact with a third set of servers that house an Structured Query Language (SQL) 2005 database. Figure 7.1 provides an example of this concept.

This is a fairly complex example, but helps illustrate the need for differing security topologies on the same network. Under no circumstances should COM+ servers or SQL 2005 servers be exposed to the Internet directly—they should be protected by placing them behind a strong security solution. At the same time, you do not want to leave IISes exposed to every hacker and script kiddie out there, so they should be placed in a DMZ or behind the first firewall or router. The idea here is to layer security so that a breach of one set of servers such as the IIS servers does not directly expose COM+ or SQL servers.

Figure 7.1 The Complex N-tier Arrangement



While differing topologies can be effectively used together, in some instances they need to be used completely separately from each other. The next sections examine the concept of security zones, how to employ them on a network, how they work, and what they can provide in regards to increased security.

Security Zones

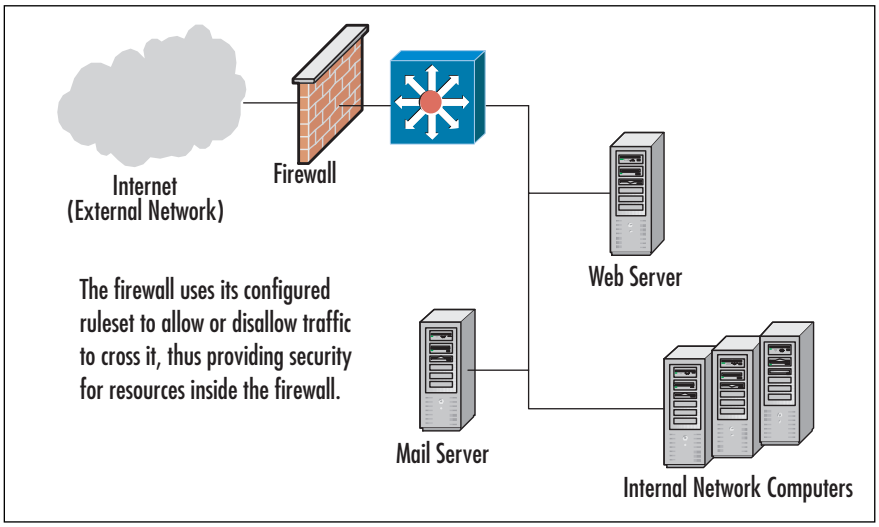
The easiest way to think of security zones is to imagine them as discrete network segments holding systems that share common requirements. These common requirements can be:

- The types of information they handle
- Who uses them
- What levels of security they require to protect their data

It is possible to have systems in a zone running different OSes, such as Windows Vista and NetWare 6.5. The type of computer, whether a PC, server, or mainframe, is not as important as the security needs of the computer. For example, there is a network that uses Windows 2003 Servers as domain controllers, Domain Name System (DNS) servers, and Dynamic Host Control Protocol (DHCP) servers. There are also Windows XP Professional clients and NetWare 6.5 file servers on the network. Some users may be using Macintosh computers running OS X or OS 9, while others may be running one or more types of Linux or UNIX. This is an extremely varied network, but it may still only have one or two security zones. As stated earlier, the type (or OS) of a computer is not as important with regards to security zones and its role.

In the early days of business Internet connectivity, the concept of security zones was developed to separate systems available to the public Internet from private systems available for internal use by an organization. A device that acted as a firewall separated the zones. Figure 7.2 shows a visual representation of the basic firewall concept.

Figure 7.2 A Basic Firewall Installation



Many of these early firewalls had only basic abilities and usually functioned only as a packet filter. Packet filters rely on ACL's. ACL's allow the packet filter to be configured to block or allow traffic based on attributes such as IP address and source and destination port. Packet filters are considered stateless, while more advanced modern firewalls like Microsoft's ISA server is considered stateful. Regardless of what type of firewall you are working with, most provide the ability to:

- Block traffic based on certain rules. The rules can block unwanted, unsolicited, spurious, or malicious traffic. (See Figure 7.3)
- Mask the presence of networks or hosts to the outside world. Firewalls can also ensure that unnecessary information about the makeup of the internal network is not available to the outside world.
- Log and maintain audit trails of incoming and outgoing traffic.
- Provide additional authentication methods.

Some newer firewalls include more advanced features, such as integrated virtual private networking (VPN) applications that allow remote users to access local systems through a secure, encrypted tunnel. Some firewalls have integrated IDSes in their product and can make firewall rule changes based on the detection of suspicious events happening at the network gateway. (IDS products and their use are covered later in this chapter.) These new technologies have much promise and make great choices for creating a "defense in depth" strategy, but remember that the more work the firewall is doing to support these other functions, the more chance there is that these additional tools may impact the throughput of the firewall device.

Figure **7.3** A Sample Firewall Rule Set

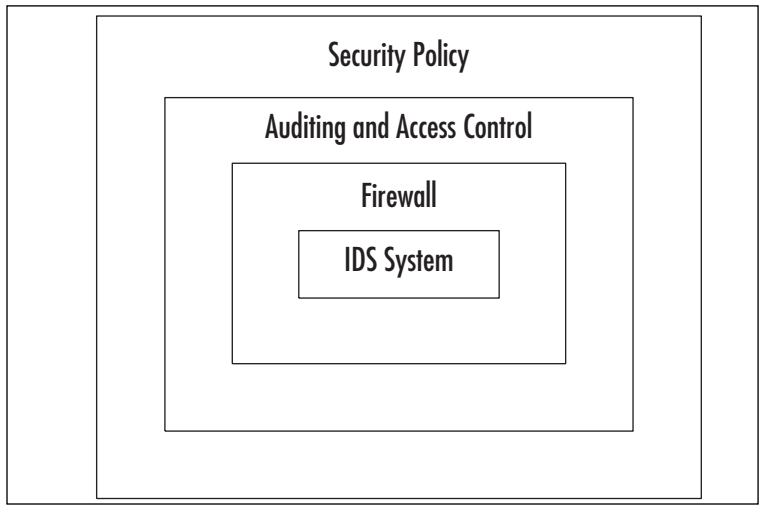
NO.	SOURCE	DESTINATION	SERVICE	ACTION	TRACK	INSTALL ON	TIME	COMMENT
-	Trusted hosts	FW1 host	FireWall1	accept	None	Gateways	Any	Enable FW1 control conne
-	ftp server	local client	expected	accept	None	Gateways	Any	Enable Response of FTP I
-	Any	Any	passive f	accept	None	Gateways	Any	Enable ftppasv connectio
-	Any	Any	rpc contr	accept	None	Gateways	Any	Enable RPC Control
1	Any	Any	Silent_Se	Drop	None	Gateways	Any	Silent drop for broadcast

Designing & Planning…

Using a Defense-in-Depth Strategy

The defense-in-depth strategy specifies the use of multiple layers of network security. In this way, you avoid depending on one single protective measure deployed on your network. In other words, to eliminate the false feeling of security because you implemented a firewall on your Internet connection, you should implement other security measures such as an IDS, auditing, and biometrics for access control. You need many levels of security (hence, defense in depth) to be able to feel safe from potential threats. A possible defense-in-depth matrix with auditing included could look like the graphic in Figure 7.4.

Figure 7.4 A Graphical Representation of Defense in Depth



In addition, when a number of these features are implemented on any single device (especially a firewall), it creates a wide opportunity for a successful attacker if that device is ever compromised. If one of these new hybrid information security devices are chosen, it is important to stay extra vigilant about applying patches and to include in the risk mitigation planning how to deal with a situation in which this device falls under the control of an attacker.

Although the installation of a firewall or hybrid device protects the internal systems of an organization, it does nothing to protect the systems that are made available to the public Internet. A different type of implementation is needed to add basic protection for those systems that are offered for public use. Thus enters the concept of the DMZ.

NOTE

A DMZ is a special section of the network, usually closest to the Internet, which uses switches, routers, and firewalls to allow access to public resources without allowing this traffic to reach the resources and computers in the private network.

Introducing the Demilitarized Zone

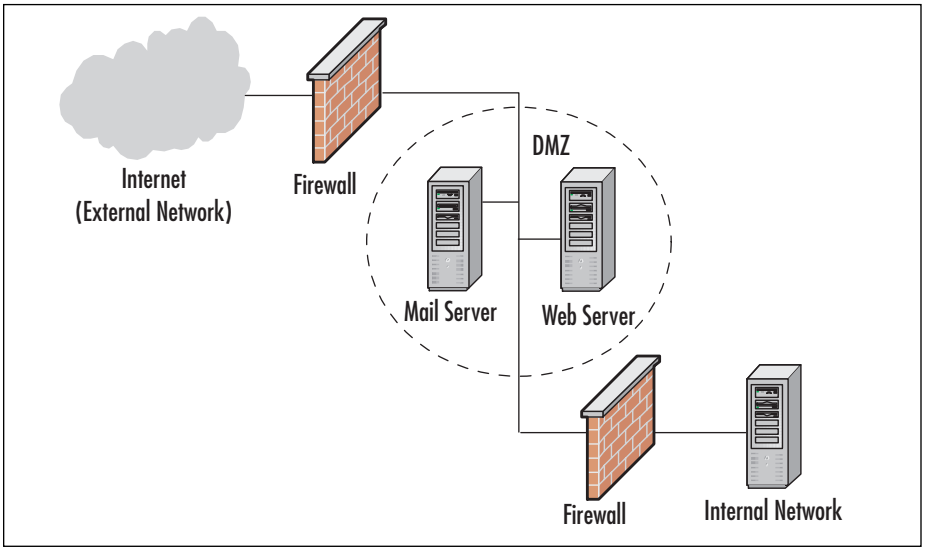
In computer security, the DMZ is a "neutral" network segment where systems accessible to the public Internet are housed, which offers some basic levels of protection against attacks. The term "DMZ" is derived from the military and is used to describe a "safe" or buffer area between two countries where, by mutual agreement, no troops or war-making activities are allowed. There are usually strict rules regarding what is allowed within the zone. When applying this term to the IT security realm, it can be used to create DMZ segments in usually one of two ways:

- Layered DMZ implementation
- Multiple interface firewall implementation

In the first method, the systems are placed between two firewall devices with different rule sets, which allows systems on the Internet to connect to the offered services on the DMZ systems, but prevents them from connecting to the computers on the internal segments of the organization's network (often called the *protected network*). Figure 7.5 shows a common installation using this layered approach.

The second method is to add a third interface to the firewall and place the DMZ systems on that network segment. (See Figure 7.6) As an example, this is the way Cisco PIX firewalls are designed. This design allows the same firewall to manage the traffic between the Internet, the DMZ, and the protected network. Using one firewall instead of two lowers the costs of the hardware and centralizes the rule sets for the network, making it easier to manage and troubleshoot problems. Currently, this multiple interface design is the preferred method for creating a DMZ segment.

Figure 7.5 A Layered DMZ Implementation

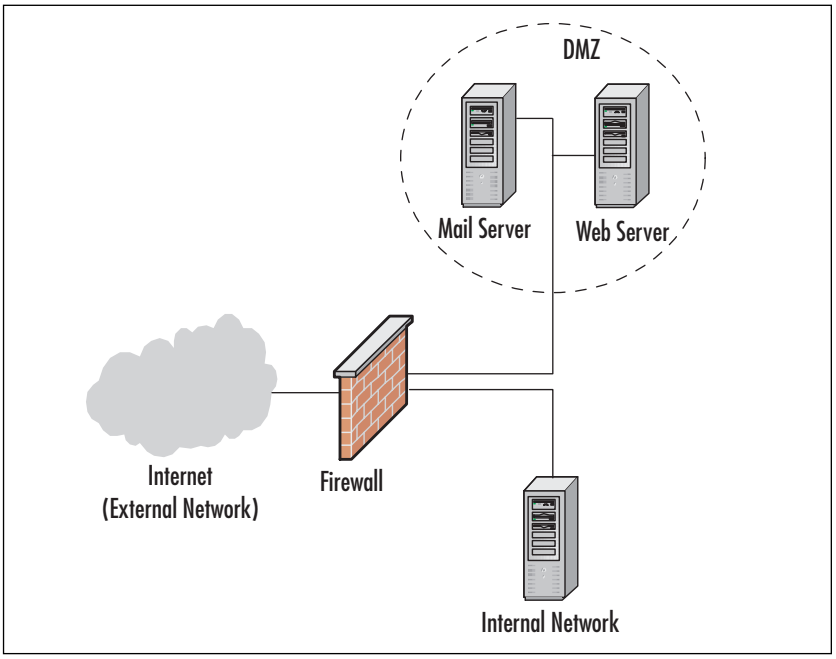


In either case, the DMZ systems are offered some level of protection from the public Internet while they remain accessible for the specific services they provide to external users. In addition, the internal network is protected by a firewall from both the external network and the systems in the DMZ. Because the DMZ systems still offer public access, they are more prone to compromise and thus they are not trusted by the systems in the protected network. A good first step in building a strong defense is to harden the DMZ systems by removing all unnecessary services and unneeded components. The result is a *bastion host*. This scenario allows for public services while still maintaining a degree of protection against attack.

NOTE

Hosts located in a DMZ are generally accessed from both internal network clients and public (external) Internet clients. Examples of DMZ bastion hosts are DNS servers, Web servers, and File Transfer Protocol (FTP) servers. A bastion host is a system on the public side of the firewall, which is exposed to attack. The word *bastion* comes from sixteenth century French word, meaning the projecting part of a fortress wall that faces the outside and is exposed to attackers.

Figure 7.6 A Multiple Interface Firewall DMZ Implementation



The role of the firewall in all of these scenarios is to manage the traffic between the network segments. The basic idea is that other systems on the Internet are allowed to access only the services of the DMZ systems that have been made public. If an Internet system attempts to connect to a service not made public, the firewall drops the traffic and logs the information about the attempt (if configured to do so). Systems on a protected network are allowed to access the Internet as they require, and they may also access the DMZ systems for managing the computers, gathering data, or updating content. In this way, systems are exposed only to attacks against the services that they offer, and not to underlying processes that may be running on them.

The systems in the DMZ can host any or all of the following services:

- **Internet Web Site Access** IIS or Apache servers that provide Web sites for public and private usage. Examples would be www.microsoft.com or www.netserverworld.com. Both of these Web sites have both publicly and privately available contents.
- **FTP Services** FTP file servers that provide public and private downloading and uploading of files. Examples would be the FTP servers used by popular download providers at www.downloads.comor www.tucows.com. FTP is designed for faster file transfer with less overhead, but does not have all of the special features that are available in Hypertext Transfer Protocol (HTTP), the protocol used for Web page transfer.
- **E-mail Relaying** A special e-mail server that acts as a middleman of sorts. Instead of e-mail passing directly from the source server to the destination server (or the next hop in the path), it passes through an e-mail relay that then forwards it. E-mail relays are a double-

edged sword and most security professionals prefer to have this function disabled on all publicly accessible e-mail servers. On the other hand, some companies have started offering e-mail relaying services to organizations as a means of providing e-mail security.

- **DNS Services** A DNS server might be placed in the DMZ in order to point incoming access requests to the appropriate server with the DMZ. This can alternatively be provided by the Internet Service Provider (ISP), usually for a nominal extra service charge. If DNS servers are placed in the DMZ, it is important to be careful and ensure that they cannot be made to conduct a zone transfer (a complete transfer of all DNS zone information from one server to another) to any server. This is a common security hole found in many publicly accessible DNS servers. Attackers typically look for this vulnerability by scanning to see if port TCP 53 is open.
- **Intrusion Detection** The placement of an IDS system (discussed later in this chapter) in the DMZ is difficult and depends on the network requirements. IDSes placed in the DMZ will tend to give more false positive results than those inside the private internal network, due to the nature of Internet traffic and the large number of script kiddies out there. Still, placing an IDS on the DMZ can give administrators early warning of attacks taking place on their network resources.

The rise of e-commerce and the increased demand of online transactions has increased the need for secure architectures and well-designed DMZ's. E-commerce requires more attention to be paid to securing transaction information that flows between consumers and the sites they use, as well as between e-commerce businesses themselves. Customer names, addresses, order information, and especially financial data need greater care and handling to prevent unauthorized access. This greater care is accomplished through the creation of the specialized segments mentioned earlier (which are similar to the DMZ) called *security zones.* Other items such as the use of encryption and the use of secure protocols like secure sockets layer (SSL) and transport layer security (TLS), are also important when designing a more secure architecture.

Multiple Needs Equals Multiple Zones

Security requirements for storing customer information and financial data are different from the requirements for storing routine, less sensitive information that businesses handle. Because this data requires processing and much of the processing is done over the Internet, more complicated network structures must be created. Many organizations choose to implement a multiple segment structure to better manage and secure their different types of business information.

This multi-segment approach allows flexibility, because new segments with specific purposes and security requirements can be easily added to the model. In general, the two segments that are widely accepted are:

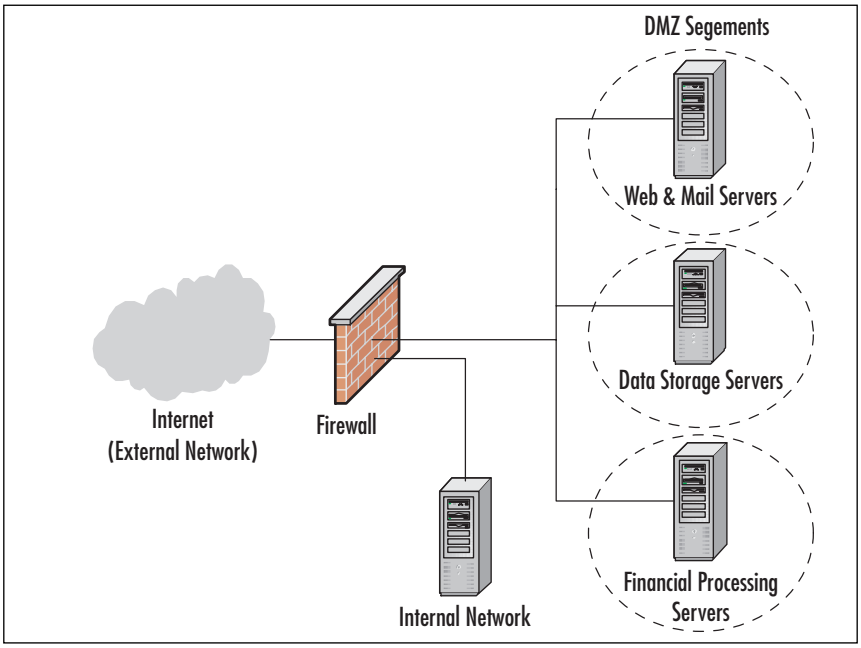
- A segment dedicated to information storage
- A segment specifically for the processing of business information

Each of these two new segments has special security and operability concerns above and beyond those of the rest of the organizational Intranet. In reality, everything comes down to dollars—what is it going to cost to implement a security solution versus what will it cost if the system is breached by

attackers. Thus the value of raw data is different than the value of the financial processing system. Each possible solution has its pluses and minuses, but in the end a balance is struck between cost versus expected results. Thus, the creation of different zones (segments) for different purposes. Note that the Web and e-mail servers would likely receive the least amount of spending and security measures, which is not to say that they will be completely ignored, they just would not receive as much as the financial servers might.

Creation of multiple segments changes a network structure to look like the drawing in Figure 7.7.

Figure 7.7 A Modern E-commerce Implementation



The diagram shown in Figure 7.7 includes the following two new zones:

- The data storage network
- The financial processing network

The *data storage zone* is used to hold information that the e-commerce application requires, such as inventory databases, pricing information, ordering details, and other non-financial data. The Web servers in the DMZ segment serve as the interface to the customers; they access the servers in the other two segments to gather the required information and to process the users' requests.

When an order is placed, the business information in these databases is updated to reflect the real-time sales and orders of the public. These business-sensitive database systems are protected from the Internet by the firewall, and they are restricted from general access by most of the systems in the protected network. This helps protect the database information from unauthorized access by an insider or from accidental modification by an inexperienced user.

The financial information from an order is transferred to the *financial processing segment*. Here, the systems validate the customer's information and then process the payment requests to a credit card company, a bank, or a transaction clearinghouse. After the information has been processed, it is stored in the database for batch transfer into the protected network, or it is transferred in real time, depending on the setup. The financial segment is also protected from the Internet by the firewall, as well as from all other segments in the setup. This system of processing the data in a location separate from the user interface creates another layer that an attacker must penetrate to gather financial information about customers. In addition, the firewall protects the financial systems from access by all but specifically authorized users inside a company.

Access controls also regulate the way network communications are initiated. For example, if a financial network system can process credit information in a store-and-forward mode, it can batch those details for retrieval by a system from the protected network. To manage this situation, the firewall permits only systems from the protected network to initiate connections with the financial segment. This prevents an attacker from being able to directly access the protected network in the event of a compromise. On the other hand, if the financial system must use real-time transmissions or data from the computers on the protected network, the financial systems have to be able to initiate those communications. In this event, if a compromise occurs, the attacker can use the financial systems to attack the protected network through those same channels. It is always preferable that DMZ systems not initiate connections into more secure areas, but that systems with higher security requirements initiate those network connections. Keep this in mind as you design your network segments and the processes that drive your site.

In large installations, these segments may vary in placement, number, and/or implementation, but this serves to generally illustrate the ideas behind the process. An actual implementation may vary from this design. For example, an administrator may wish to place all the financial processing systems on the protected network. This is acceptable as long as the requisite security tools are in place to adequately secure the information. I have also seen implementation of the business information off an extension of the DMZ, as well as discrete DMZ segments for development and testing. Specific technical requirements will impact actual deployment, so administrators may find that what they currently have in place on a network (or the need for a future solution) may deviate from the diagrams shown earlier. The bottom line is to ensure that systems are protected.

Problems with Multi-zone Networks

Some common problems do exist with multiple-zone networks. By their very nature they are complex to implement, protect, and manage. Firewall rule sets are often large, dynamic, and confusing, and the implementation can be arduous and resource intensive.

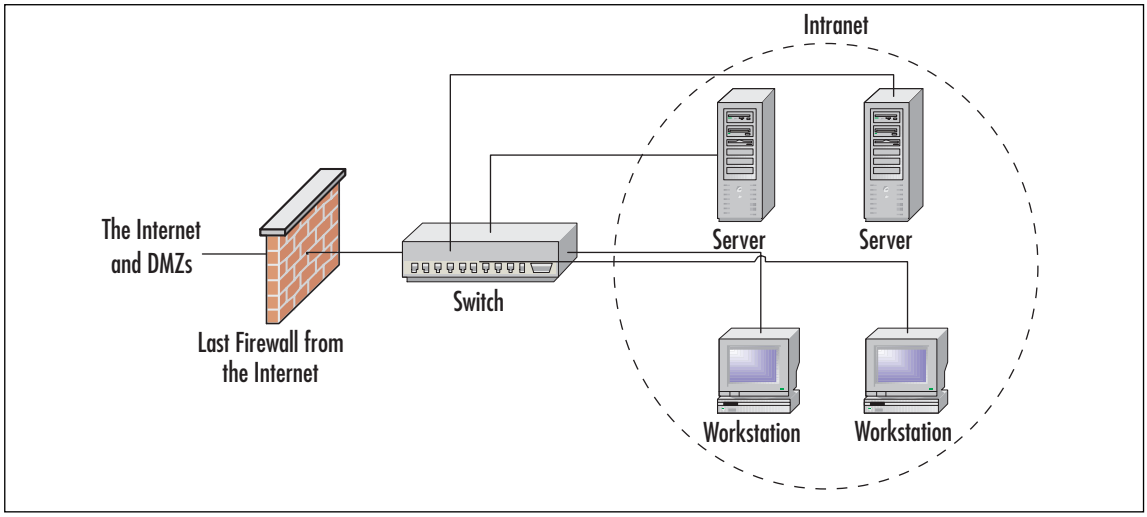
Creating and managing security controls such as firewall rules, IDS signatures, and user access regulations is a large task. These processes should be kept as simple as possible without compromising security or usability. It is best to start with deny-all strategies and permit only the services and network transactions required to make the site function, and then carefully manage the site's performance making small changes to the access controls to more easily manage the rule sets. Using these guidelines, administrators should be able to quickly get the site up and running without creating obvious security holes in the systems.

As a site grows and offers new features, new zones may have to be created. The above process should be repeated for creating the rule sets governing these new segments. As always, it is important to audit and inspect any changes and keep backups of the old rule sets in case they are needed again.

Intranet

Thus far, this chapter has only discussed the systems that reside outside of the protected internal network. These servers are the ones that are located in the DMZ. The rest of the internal network is called the intranet, which means a private internal network. The intranet, therefore, is every part of a network that lies on the inside of the last firewall from the Internet. Figure 7.8 gives an example of an intranet.

Figure 7.8 A Simple Intranet Example



It is expected that all traffic on the intranet will be secure and safe from the prying eyes on the Internet. It is the network security professional's job to make sure that this happens. While a security breach of a DMZ system can be costly to a company, a breach that occurs inside an intranet could be *extraordinarily* costly and damaging. If this happens, customers and business partners might lose faith in the company's ability to safeguard sensitive information, and other attackers will likely make the network a favorite target for future attacks.

To ensure that all traffic on the intranet is secure, the following issues should be addressed:

- Make sure that the firewall is configured properly to stop attack attempts at the firewall. There are many different opinions on how to do this, but the majority of security professionals agree that you should start with a deny all or "block everything" mentality and then open the firewall on a case-by-case basis, thereby only allowing specific types of traffic to cross it (regardless of which direction the traffic is flowing). It's important to remember that each open port and service offers the attacker an additional path from which he may potentially target the network.

- Additionally, make sure that the firewall is configured properly to prevent unauthorized network traffic, such as file sharing programs (for example, BitTorrent, Gnutella or Morpheus) from being used on the internal network. These types of programs can sometimes be difficult to block, but it can be done.
- Make sure the firewall will watch traffic that egresses or leaves the network from trusted hosts, and ensure that it is not intercepted and altered en route; steps should also be taken to try to eliminate spoofing from attackers.
- Make sure that the antivirus software is in use and up to date. Consider implementing an enterprise-level solution, consisting of a central server responsible for coordinating and controlling the identification and collection of viruses on your network.
- Educate users on the necessity of keeping their computers logged out when not in use.
- Implement Secure Internet Protocol (IPSec) on the intranet between all clients and servers to prevent eavesdropping; note that more often than not, the greatest enemy lies on the inside of the firewall.
- Conduct regular, but unannounced, security audits and inspections. Be sure to closely monitor all logs that are applicable.
- Do not allow the installation of modems or unsecured wireless access points on any intranet computers. Do not allow any connection to the Internet except through the firewall and proxy servers, as applicable.

NOTE

A *proxy server* is a server that sits between an intranet and its Internet connection. Proxy servers provide features such as document caching (for faster browser retrieval) and access control. Proxy servers can provide security for a network by filtering and discarding requests that are deemed inappropriate by an administrator. Proxy servers also protect the internal network by masking all internal IP addresses—all connections to Internet servers appear to be coming from the IP address of the proxy servers.

Of course, there are literally hundreds of other issues that may need to be addressed but these are some of the easiest ones to take care of and the most commonly exploited ones.

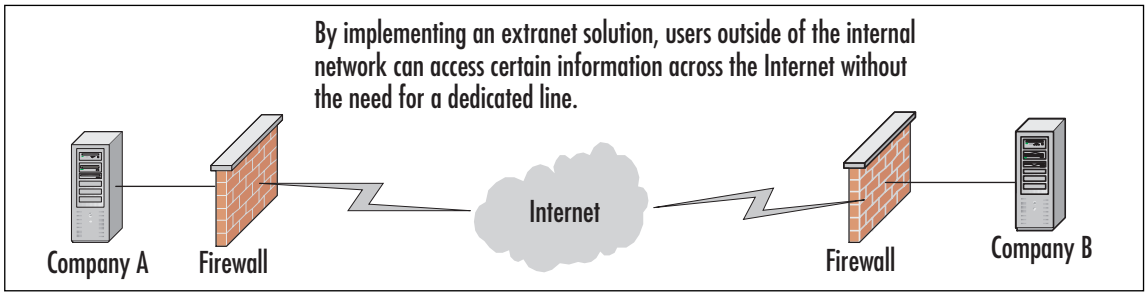
Extranet

Extranets are a special implementation of the intranet topology. Creating an extranet allows for access to a network (more likely, certain parts of a network) by trusted customers, partners, or other users. These users, who are external to the network—they are on the Internet side of the firewalls and other security mechanisms—can then be allowed to access private information stored on the internal network that they would not want to place on the DMZ for general public access. The amount of

access that each user or group of users is allowed to have to the intranet can be easily customized to ensure that each user or group gets what they need and nothing more. Additionally, some organizations create extranets to allow their own employees to have access to certain internal data while away from the private network.

The following is an example of how two companies might each choose to implement an extranet solution for their mutual betterment. Company A makes door stoppers and has recently entered into a joint agreement with Company B. Company B makes cardboard boxes. By partnering together, both companies are hoping to achieve some form of financial gain. Company A is now able to get cardboard boxes (which it needs to ship its product) made faster, cheaper, and to exact specification; Company B benefits from newfound revenue from Company A. Everybody wins and both companies are very happy. After some time, both companies realize that they could streamline this process even more if they each had access to certain pieces of data about the other company. For example, Company A wants to keep track of when its cardboard boxes will be arriving. Company B, on the other hand, wants to be able to control box production by looking at how many orders for door stoppers Company A has. What these two companies need is an extranet. By implementing an extranet solution, both companies will be able to get the specific data they need to make their relationship even more profitable, without either company having to grant full, unrestricted access to its private internal network. Figure 7.9 graphically depicts this extranet solution.

Figure 7.9 A Simple Extranet Example



Users attempting to gain access to an extranet require some form of authentication before they are allowed access to resources. The type of access control implemented can vary, but some of the more common include usernames/passwords and digital certificates. Once an extranet user has been successfully authenticated, they can gain access to the resources that are allowed for their access level. In the previous example, a user from Company B's production department might need to see information about the number of door stoppers being ordered, while a user from Company A's shipping department might need to see information detailing when the next shipment of boxes is expected.

VLANs

A VLAN can be thought of as the equivalent to a *broadcast domain*.

NOTE

A broadcast domain consists of a group of nodes (computers) that receive layer 2 broadcasts sent by other members of the same group. Typically, broadcast domains are separated by creating additional network segments or by adding a router.

Do not confuse broadcast domains with collision domains. Collision domains refer specifically to Ethernet networks. The area of network cabling between layer 2 devices is known as a *collision domain*. Layer 2 devices typically include switches that rely on the physical address (Media Access Control [MAC] address) of computers to route traffic.

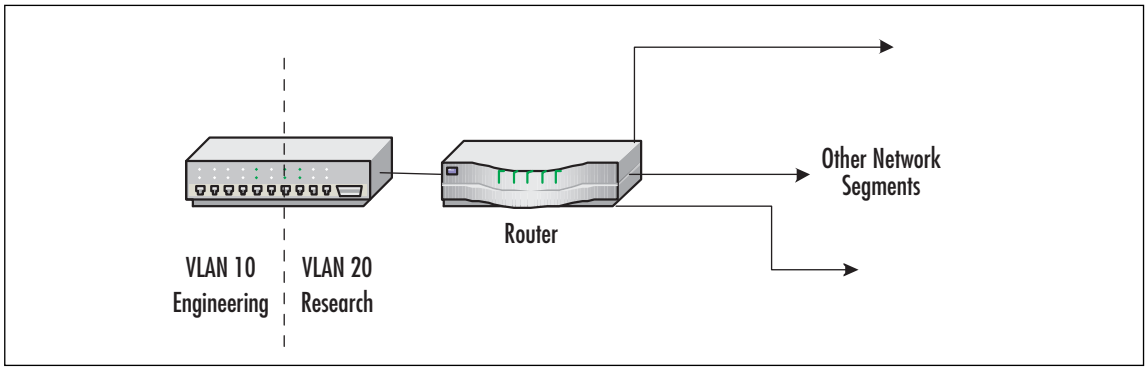
VLANs are a way to segment a network, as discussed above. When thinking of a VLAN, think of taking a switch and physically cutting it into two or more pieces with an axe. Special software features found in newer, more expensive switches, allow administrators to physically split one physical switch into multiple logical switches, thus creating multiple network segments that are completely separate from one another.

The VLAN is thus a *logical* local area network that uses a basis other than a physical location to map the computers that belong to each separate VLAN (e.g., each department within a company could comprise a separate VLAN, regardless of whether or not the department's users are located in physical proximity). This allows administrators to manage these virtual networks individually for security and ease of configuration.

Let's look at an example of using VLANs. There is an Engineering section consisting of 14 computers and a Research section consisting of 8 computers, all on the same physical subnet. Users typically communicate only with other systems within their respective sections. Both sections share the use of one Cisco Catalyst 2924 XL switch. To diminish the size of the necessary broadcast domain for each section, the administrator can create two VLANs, one for the Engineering section and one for the Research section. After creating the two VLANs, all broadcast traffic for each section will be isolated to its respective VLAN. But what happens when a node in the Engineering section needs to communicate with a node in the Research section? Do the two systems connect from within the Catalyst 2924 XL switch? No; this cannot occur since the two sections have been set up on two different VLANs. For traffic to be passed between VLANs (even when they are on the same switch) a router must be used.

Figure 7.10 graphically depicts the previous example of splitting one switch into two VLANs. Note that two switches can also be split into two VLANs or more, depending on the need. The following example shows how to split two switches into multiple VLANs with each VLAN acting as its own physically separated network segment. In reality, many more VLANs can be created; they are only limited by port density (the number of ports on a switch) and the feature set of the switch's software.

Figure 7.10 Using VLANs to Segment Network Traffic



Each VLAN functions like a separate switch due to the combination of hardware and software features built into the switch itself. Thus, the switch must be capable of supporting VLANs in order to use them. The following are typical characteristics of VLANs when implemented on a network:

- Each VLAN is the equivalent of a physically separate switch as far as network traffic is concerned.
- A VLAN can span multiple switches, limited only by imagination and the capabilities of the switches being used.
- Trunks carry the traffic between each switch that is part of a VLAN. A trunk is defined as a point-to-point link from one switch to another switch. The purpose of a trunk is to carry the traffic of multiple VLANs over a single link.
- Cisco switches, for example, use the Cisco proprietary Inter-Switch Link (ISL) and IEEE 802.1Q protocol as their trunking protocols.

Network Address Translation

NAT was developed because of the explosive growth of the Internet and the increase in home and business networks—the number of available IP addresses was simply not enough. A computer must have an IP address in order to communicate with other computers on the Internet. NAT allows a single device, such as a router, to act as an agent between the Internet and the local network. This device or router provides a pool of addresses to be used by your local network. Only a single, unique IP address is required to represent this entire group of computers. The outside world is unaware of this division and thinks that only one computer is connected. Common types of NAT include:

- **Static NAT** Used by businesses to connect Web servers to the Internet
- **Dynamic NAT** Larger business use this type of NAT because it can operate with a pool of public addresses
- **Port Address Translation (PAT)** Most home networks using Digital Subscriber Line (DSL) or cable modems use this type of NAT

NAT is a feature of many routers, firewalls, and proxies. NAT has several benefits, one of which is its ability to hide the IP address and network design of the internal network. The ability to hide the internal network from the Internet reduces the risk of intruders gleaning information about the network and exploiting that information to gain access. If an intruder does not know the structure of a network, the network layout, the names and IP address of systems, and so on, it is very difficult to gain access to that network. NAT enables internal clients to use nonroutable IP addresses, such as the private IP addresses defined in RFC 1918, but still enables them to access Internet resources. The three ranges of IP addresses RFC 1918 reserved includes:

```
10.0.0.0 - 10.255.255.255 (10/8 prefix)
172.16.0.0 - 172.31.255.255 (172.16/12 prefix)
192.168.0.0 - 192.168.255.255 (192.168/16 prefix)
```

NAT can be used when there are many internal private IP addresses and there are only a few public IP addresses available to the organization. In this situation, the company can share the few public IP addresses among all the internal clients. NAT can also aid in security as outsiders cannot directly see Internal IP addresses. Finally, NAT restricts traffic flow so that only traffic requested or initiated by an internal client can cross the NAT system from external networks.

When using NAT, the internal addresses are reassigned to private IP addresses and the internal network is identified on the NAT host system. Once NAT is configured, external malicious users are only able to access the IP address of the NAT host that is directly connected to the Internet, but they are not able to "see" any of the internal computers that go through the NAT host to access the Internet.

Damage & Defense…

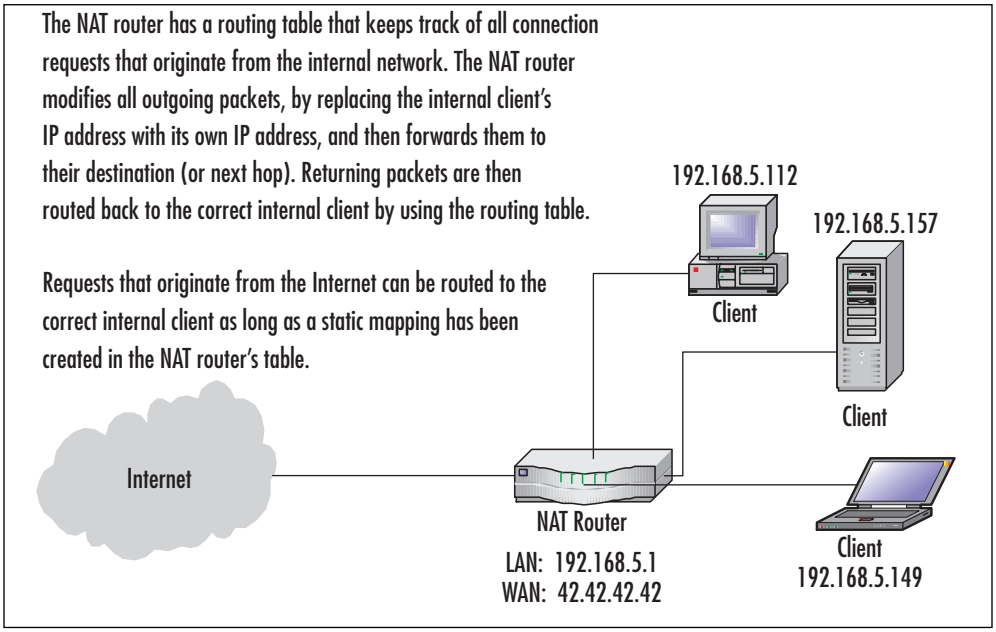
Deploying a NAT Solution

NAT is relatively easy to implement, and there are several ways to do so. Many broadband hardware devices (cable and DSL modems) are called cable/DSL "routers," because they allow you to connect multiple computers. However, they are actually combination modem/NAT devices rather than routers, because they require only one external (public) IP address. You can also buy NAT devices that attach your basic cable or DSL modem to the internal network. Alternatively, the computer that is directly connected to a broadband modem can use NAT software to act as the NAT device itself. This can be an add-on software program or the NAT software that is built into some OSes. For example, Windows XP and Vista include a fully configurable NAT as part of its Routing and Remote Access services. Even older versions of Microsoft products such as Windows 98SE, Me, and 2000 Professional include a "lite" version of NAT called Internet Connection Sharing (ICS).

For a quick, illustrated explanation of how NAT works with a broadband connection, see the HomeNetHelp article at www.homenethelp.com/web/explain/about-NAT.asp.

When NAT is used to hide internal IP addresses (see Figure 7.11), it is sometimes called a *NAT firewall*; however, do not let the word firewall give you a false sense of security. NAT by itself solves only one piece of the security perimeter puzzle. A true firewall does much more than link private IP addresses to public ones, and vice versa.

Figure 7.11 NAT Hides the Internal Addresses



Configuring & Implementing...

Public and Private Addressing

Certain IP address ranges are classified as Private IP addresses, meaning they are not to be routed on the Internet. These addresses are intended only for use on private internal networks. There are three groups of private IP addresses under the IPv4 standard as outlined here:

```
10.0.0.0 - 10.255.255.255 (10/8 prefix)
172.16.0.0 - 172.31.255.255 (172.16/12 prefix)
192.168.0.0 - 192.168.255.255 (192.168/16 prefix)
```

The network segment shown in Figure 7.11 uses private IP addresses on the internal network from the 192.168.5.x subnet. The allowable addresses in this subnet would then be 192.168.5.1 through 192.168.5.254. The 192.168.5.255 address is con-

Continued

sidered to be a broadcast address—one that would be used if a computer needed to send a transmission to all other computers on that subnet. Typically, the gateway or router will occupy the first address in a given range (as is the case in Figure 7.11), where the router has been assigned the address of 192.168.5.1 on its LAN interface.

Note that in Exercise 7.01, the ICS host computer is statically assigned the IP address 192.168.0.1 and all ICS clients will automatically be assigned IP addresses in the 192.168.0.x range so that they can communicate directly with the ICS host without needing a router.

For a complete discussion on private IP addresses, see RFC 1918 at ftp://ftp.rfc-editor.org/in-notes/rfc1918.txt. The Internet Assigned Numbers Authority (IANA) maintains a current listing of all IPv4 IP address range assignments at www.iana.org/assignments/ipv4-address-space. You can also examine all of the special IPv4 IP address assignments at ftp://ftp.rfc-editor.org/in-notes/rfc3330.txt.

Tunneling

Tunneling is used to create a virtual tunnel (a virtual point-to-point link) between you and your destination using an untrusted public network as the medium. In most cases, this would be the Internet. When establishing a tunnel, commonly called a VPN, a safe connection is being created between two points that cannot be examined by outsiders. In other words, all traffic that is traveling through this tunnel can be seen but cannot be understood by those on the outside. All packets are encrypted and carry information designed to provide authentication and integrity. This ensures that they are tamper-proof and thus can withstand common IP attacks, such as the Man-in—Middle (MITM) and packet replay. When a VPN is created, traffic is private and safe from prying eyes.

NOTE

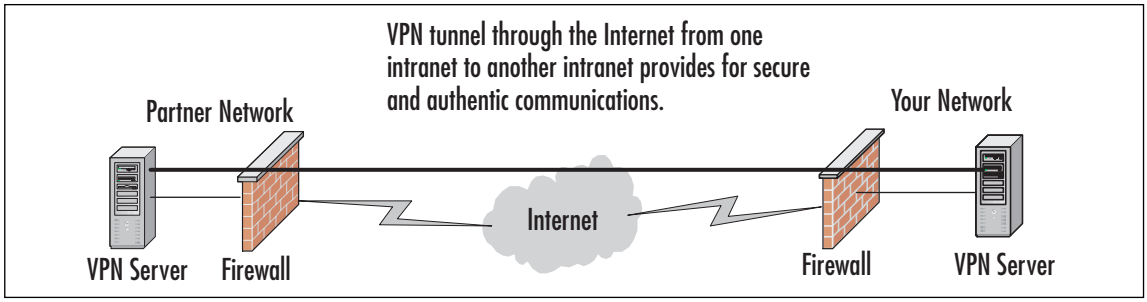
Tunneling is used in conjunction with encryption to provide total end-to-end data protection across an untrustworthy network, such as the Internet. Point-to-Point Tunneling Protocol (PPTP) and Layer 2 Tunneling Protocol (L2TP) are popular VPN tunneling protocols, while Microsoft Point-to-Point Encryption (MPPE) and IPSec are their encryption counterparts. Do not confuse tunneling with encryption.

VPN tunneling provides confidentiality of data, in that the traffic is encrypted, typically using MPPE or IPSec. VPNs created using the L2TP use IPSec for encryption, whereas tunnels created with the PPTP use MPPE. Windows XP and newer Microsoft OSes can use IPSec; all older versions must use MPPE.

Most other new OSes also provide support for L2TP and IPSec. Tunnels can also be created using IPSec alone (without L2TP) or using Secure Shell (SSH) or Crypto Internet Protocol Encapsulation (CIPE) in Linux/UNIX environments. It is important to understand that tunneling and encryption are two separate processes, both of which are necessary to create a VPN.

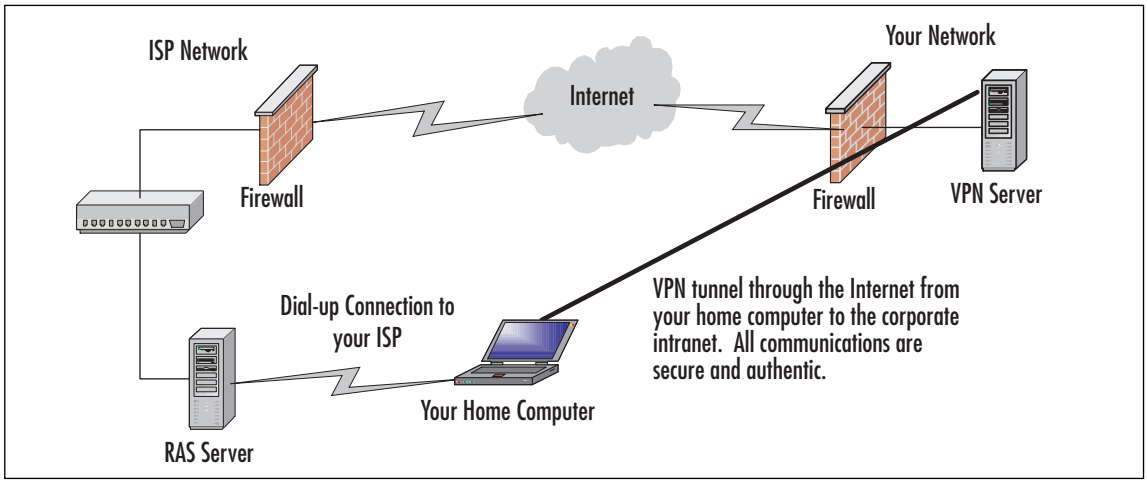
For more information about VPN technologies, see http://en.wikipedia.org/wiki/VPN. Tunneling is often used when configuring and implementing an extranet solution, but is not limited to usage only in that situation. Consider Figure 7.12, where we have created a VPN tunnel from your network to the network of a business partner.

Figure 7.12 Setting Up a Business-to-business VPN



You can also establish a VPN from your home computer to the corporate network by making use of your ISP connection, as shown in Figure 7.13.

Figure 7.13 Establishing a VPN Tunnel to Access the Corporate Network from Home



Intrusion Detection

Firewalls and other simple boundary devices lack some degree of intelligence when it comes to observing, recognizing, and identifying attack signatures that may be present in the traffic they monitor and the log files they collect. A successful security strategy requires many layers and components. One of these components is the IDS. Intrusion detection is an important piece of security in that it acts as a detective control. As an example, consider a locked car in a parking lot. Locking the car is

much like securing the network. It provides security but only deters attacks. What if someone breaks in the locked car, how would the driver detect this? In the world of automobile security that could be accomplished with an alarm system. In the computer world this is done with an IDS. Whereas other boundary devices may collect all the information necessary to detect (and often to foil) attacks that may be getting started or are already underway, they have not been programmed to inspect for and detect the kinds of traffic or network behavior patterns that match known attack signatures or that suggest potential unrecognized attacks may be incipient or in progress.

In a nutshell, the simplest way to define an IDS is to describe it as a specialized tool that knows how to read and interpret the contents of log files from sensors placed on the network, routers, firewalls, servers, and other network devices. Furthermore, an IDS often stores a database of known attack signatures and can compare patterns of activity, traffic, or behavior it sees in the logs it is monitoring against those signatures to recognize when a close match between a signature and current or recent behavior occurs. At that point, the IDS can issue alarms or alerts, take various kinds of automatic action ranging from shutting down Internet links or specific servers to launching backtraces, and make other active attempts to identify attackers and actively collect evidence of their nefarious activities.

By analogy, an IDS does for a network what an antivirus software package does for files that enter a system: it inspects the contents of network traffic to look for and deflect possible attacks, just as an antivirus software package inspects the contents of incoming files, e-mail attachments, active Web content, and so forth to look for virus signatures (patterns that match known malicious software [malware]) or for possible malicious actions (patterns of behavior that are at least suspicious, if not downright unacceptable).

To be more specific, intrusion detection means detecting unauthorized use of or attacks on a system or network. An IDS is designed and used to detect and then to deflect or deter (if possible) such attacks or unauthorized use of systems, networks, and related resources. Like firewalls, IDSes may be software-based or may combine hardware and software (in the form of preinstalled and preconfigured standalone IDS devices). There are many opinions as to what is the best option. For the exam what's important is to understand the differences. Often, IDS software runs on the same devices or servers where firewalls, proxies, or other boundary services operate; an IDS *not* running on the same device or server where the firewall or other services are installed to monitor those devices closely and carefully. Although such devices tend to operate at network peripheries, IDS systems can detect and deal with insider attacks as well as external attacks as long as the sensors are appropriately placed to detect such attacks.

Characterizing IDSes

IDS systems vary according to a number of criteria. By explaining those criteria, we can explain what kinds of IDSes you are likely to encounter and how they do their jobs. First and foremost, it is possible to distinguish IDSes on the basis of the kinds of activities, traffic, transactions, or systems they monitor. In this case, IDSes may be divided into network-based, host-based, and application-based types. IDSes that monitor network backbones and look for attack signatures are called *network-based IDSes*, whereas those that operate on hosts defend and monitor the operating and file systems for signs of intrusion and are called *host-based IDSes.* Some IDSes monitor only specific applications and are called *application-based IDSes.* (This type of treatment is usually reserved for important applications such as database management systems, content management systems, accounting systems, and so forth.) Read on to learn more about these various types of IDS monitoring approaches:

- Network-based IDS Characteristics
  - **Pros** Network-based IDSes can monitor an entire large network with only a few well-situated nodes or devices, and impose little overhead on a network. Network-based IDSes are mostly passive devices that monitor ongoing network activity without adding significant overhead or interfering with network operation. They are easy to secure against attack and may even be undetectable to attackers; they also require little effort to install and use on existing networks.
  - **Cons** Network-based IDSes may not be able to monitor and analyze all traffic on large, busy networks, and may therefore overlook attacks launched during peak traffic periods. Network-based IDSes may not be able to monitor switch-based (high-speed) networks effectively, either. Typically, network-based IDSes cannot analyze encrypted data, nor do they report whether or not attempted attacks succeed or fail. Thus, network-based IDSes require a certain amount of active, manual involvement from network administrators to gauge the effects of reported attacks.
- Host-based IDS Characteristics
  - **Pros** A host-based IDS can analyze activities on the host it monitors at a high level of detail; it can often determine which processes and/or users are involved in malicious activities. Though they may each focus on a single host, many host-based IDS systems use an agent-console model where agents run on (and monitor) individual hosts, but report to a single centralized console (so that a single console can configure, manage, and consolidate data from numerous hosts). Host-based IDSes can detect attacks undetectable to the network-based IDS and can gauge attack effects quite accurately. Host-based IDSes can use host-based encryption services to examine encrypted traffic, data, storage, and activity. Host-based IDSes also have no difficulties operating on switch-based networks.
  - **Cons** Data collection occurs on a per-host basis; writing to logs or reporting activity requires network traffic and can decrease network performance. Clever attackers who compromise a host can also attack and disable host-based IDSes. Host-based IDSes can be foiled by Denial of Service (DoS) attacks, because they may prevent any traffic from reaching the host where they are running or prevent reporting on such attacks to a console elsewhere on a network. Most significantly, a host-based IDS consumes processing time, storage, memory, and other resources on the hosts where such systems operate.
- Application-based IDS Characteristics
  - **Pros** Application-based IDSes concentrate on events occurring within some specific application. They often detect attacks through analysis of application log files and can usually identify many types of attacks or suspicious activity. Sometimes an application-based IDS can track unauthorized activity from individual users. They can also work with encrypted data, using application-based encryption/decryption services.
  - **Cons** Application-based IDSes are sometimes more vulnerable to attack than the host-based IDS. They can also consume significant application (and host) resources.

In practice, most commercial environments use some combination of network-, host-, and/or application-based IDS systems to observe what is happening on the network while also monitoring key hosts and applications more closely.

It's also important to understand that an IDS can operate in one of four states. These include:

- **Positive** An attack occurred and the IDS detected it
- **Negative** No attack occurred and none was detected
- **False Positive** No attack occurred yet the IDS believes one did and triggered an alert
- **False Negative** An attack occurred yet was not detected

As you can imagine, these states are not all the same. The goal of the security professional tuning the IDS is to configure it in such a way so that attacks are detected and false alarms do not occur. In reality, this is not always so easy as it can take a lot of time and effort to get an IDS properly set up. If configured incorrectly, there may be too many false positives so that users become desensitized and begin to ignore the alarms. There is even a worse condition in that the IDS may be misconfigured so that false negatives occur. In this condition, an attack that has happened may never be detected.

IDSes may also be distinguished by their differing approaches to event analysis. Some IDSes primarily use a technique called *signature detection*. This resembles the way many antivirus programs use virus signatures to recognize and block infected files, programs, or active Web content from entering a computer system, except that it uses a database of traffic or activity patterns related to known attacks, called *attack signatures*. Indeed, signature detection is the most widely used approach in commercial IDS technology today. Another approach is called *anomaly detection*, which uses rules or predefined concepts about "normal" and "abnormal" system activity (called *heuristics*) to distinguish anomalies from normal system behavior and to monitor, report on, or block anomalies as they occur. Some IDSes support limited types of anomaly detection; most experts believe this kind of capability will become part of how more IDSes operate in the future. Read on for more information about these two kinds of event analysis techniques:

- Signature-based IDS characteristics
  - **Pros** A signature-based IDS examines ongoing traffic, activity, transactions, or behavior for matches with known patterns of events specific to known attacks. As with antivirus software, a signature-based IDS requires access to a current database of attack signatures and some way to actively compare and match current behavior against a large collection of signatures. Except when entirely new, uncataloged attacks occur, this technique works extremely well.
  - **Cons** Signature databases must be constantly updated, and IDSes must be able to compare and match activities against large collections of attack signatures. If signature definitions are too specific, a signature-based IDS may miss variations on known attacks. (A common technique for creating new attacks is to change existing known attacks rather than to create entirely new ones from scratch.) Signature-based IDSes can also impose noticeable performance drags on systems when current behavior matches multiple (or numerous) attack signatures, either in whole or in part.

- Anomaly-based IDS characteristics
  - **Pros** An anomaly-based IDS examines ongoing traffic, activity, transactions, or behavior for anomalies on networks or systems that may indicate attack. The underlying principle is the notion that "attack behavior" differs enough from "normal user behavior" that it can be detected by cataloging and identifying the differences involved. By creating baselines of normal behavior, anomaly-based IDS systems can observe when current behavior deviates statistically from the norm. This capability theoretically gives anomaly-based IDSes the ability to detect new attacks that are neither known nor for which signatures have been created.
  - **Cons** Because normal behavior can change easily and readily, anomaly-based IDS systems are prone to false positives, where attacks may be reported based on changes to the norm that are "normal," rather than representing real attacks. Their intensely analytical behavior can also impose heavy processing overheads on systems they are running. Furthermore, anomaly based systems take a while to create statistically significant baselines (to separate normal behavior from anomalies); they are relatively open to attack during this period.

Today, many antivirus packages include both signature-based and anomaly based detection characteristics, but only a few IDSes incorporate both approaches. Most experts expect anomaly based detection to become more widespread in IDSes, but research and programming breakthroughs will be necessary to deliver the kind of capability that anomaly based detection should be, but is currently not able to deliver.

By implementing the following techniques, IDSes can fend off expert and novice hackers alike. Although experts are more difficult to block entirely, these techniques can slow them down considerably:

- Breaking TCP connections by injecting reset packets into attacker connections causes attacks to fall apart.
- Deploying automated packet filters to block routers or firewalls from forwarding attack packets to servers or hosts under attack stops most attacks cold—even DoS or Distributed Denial of Service (DDoS) attacks. This works for attacker addresses and for protocols or services under attack (by blocking traffic at different layers of the ARPA networking model, so to speak).
- Deploying automated disconnects for routers, firewalls, or servers can halt all activity when other measures fail to stop attackers (as in extreme DDoS attack situations, where filtering would only work effectively on the ISP side of an Internet link, if not higher up the ISP chain as close to Internet backbones as possible).
- Actively pursuing reverse DNS lookups or other ways of attempting to establish hacker identity is a technique used by some IDSes, generating reports of malicious activity to all ISPs in the routes used between the attacker and the attackee. Because such responses may themselves raise legal issues, experts recommend obtaining legal advice before repaying hackers in kind.

Signature-based IDSes and Detection Evasion

An IDS is, quite simply, the high-tech equivalent of a burglar alarm configured to monitor access points, hostile activities, and known intruders. These systems typically trigger on events by referencing network activity against an *attack signature database*. If a match is made, an alert takes place and is logged for future reference. It is the makeup of this signature database that is the Achilles heel of these systems.

Attack signatures consist of several components used to uniquely describe an attack. The signature is a kind of detailed profile that is compiled by doing an analysis of previous successful attacks. An ideal signature would be one that is specific to the attack, while being as simple as possible to match with the input data stream (large complex signatures may pose a serious processing burden). Just as there are varying types of attacks, there must be varying types of signatures. Some signatures define the characteristics of a single IP option, perhaps that of an nmap portscan, while others are derived from the actual payload of an attack.

Most signatures are constructed by running a known exploit several times, monitoring the data as it appears on the network, and looking for a unique pattern that is repeated on every execution. This method works fairly well at ensuring that the signature will consistently match an attempt by that particular exploit. Remember, the idea is for the *unique* identification of an attack, not merely the detection of attacks.

A computing system, in its most basic abstraction, can be defined as a finite state machine, which literally means that there are only a specific predefined number of states that a system may attain. This limitation hinders the IDS, in that it can be well armed at only a single point in time (in other words, as well armed as the size of its database). This poses several problems:

- First, how can one have foreknowledge of the internal characteristics that make up an intrusion attempt that has not yet occurred? You cannot alert on attacks you have never seen.
- Second, there can be only educated guesses that what has happened in the past may again transpire in the future. You can create a signature for a past attack after the fact, but that is no guarantee you will ever see that attack again.
- Third, an IDS may be incapable of discerning a new attack from the background white noise of any network. The network utilization may be too high, or many false positives cause rules to be disabled.
- And finally, the IDS may be incapacitated by even the slightest modification to a known attack. A weakness in the signature matching process, or more fundamentally, a weakness in the packet analysis engine (packet sniffing/reconstruction) will thwart any detection capability.

The goals of an attacker in relation to IDS evasion are twofold:

- To evade detection completely
- To use techniques and methods that increase the processing load of the IDS sensor significantly

As more methods are employed by attackers on a wide scale, more vendors will be forced to implement more complex signature matching and packet analysis engines. These complex systems will undoubtedly have lower operating throughputs and will present more opportunities for evasion. The paradox is that the more complex a system becomes, the more opportunities there are for vulnerabilities. Some say the ratio for bugs to code may be as high as 1:1000, and even conservatives say a ratio of 1:10000 may exist. With these sorts of figures in mind, a system of increasing complexity will undoubtedly lead to new levels of increased insecurity.

Finally, advances in IDS design have led to a new type of IDS, called an intrusion prevention system (IPS). An IPS is capable of responding to attacks when they occur. This behavior is desirable from two points of view. For one thing, a computer system can track behavior and activity in near-real time and respond much more quickly and decisively during the early stages of an attack. Since automation helps hackers mount attacks, it stands to reason that it should also help security professionals fend them off as they occur. For another thing, an IPS can stand guard and run 24 hours per day/7 days per week, but network administrators may not be able to respond as quickly during off hours as they can during peak hours. By automating a response and moving these systems from detection to prevention they actually have the ability to block incoming traffic from one or more addresses from which an attack originates. This allows the IPS the ability to halt an attack in process and block future attacks from the same address.

Popular Commercial IDS Systems

Literally hundreds of vendors offer various forms of commercial IDS implementations. The most effective solutions combine network- and host-based IDS implementations. Likewise, most such implementations are primarily signature-based, with only limited anomaly based detection capabilities present in certain specific products or solutions. Finally, most modern IDSes include some limited automatic response capabilities, but these usually concentrate on automated traffic filtering, blocking, or disconnects as a last resort. Although some systems claim to be able to launch counterstrikes against attacks, best practices indicate that automated identification and backtrace facilities are the most useful aspects that such facilities provide and are therefore those most likely to be used.

A huge number of potential vendors can provide IDS and IPS products to companies and organizations. Without specifically endorsing any particular vendor, the following products offer some of the most widely used and best-known solutions in this product space:

- **Cisco Systems** is best known for its switches and routers, but offers significant firewall and intrusion detection products as well (www.cisco.com).
- **GFI LANguard** is a family of monitoring, scanning, and file integrity check products that offer broad intrusion detection and response capabilities (www.gfi.com/languard/).
- **Internet Security Systems (ISS)** offers a family of enterprise-class security products called RealSecure, that includes comprehensive intrusion detection and response capabilities (www.iss.net).
- **McAfee** offers the IntruShield IPS systems that can handle gigabit speeds and greater (www.mcafee.com).

- **Sourcefire** is the best known vendor of open source IDS software as they are the developers of Snort, which is an open source IDS application that can be run on Windows or Linux systems (www.snort.org).

A clearinghouse for ISPs known as ISP-Planet offers all kinds of interesting information online about MSSPs, plus related firewall, VPN, intrusion detection, security monitoring, antivirus, and other security services. For more information, visit any or all of the following URLs:

- ISP-Planet Survey: Managed Security Service Providers, participating provider's chart, www.isp-planet.com/technology/mssp/participants_chart.html.
- Managed firewall services chart, www.isp-planet.com/technology/mssp/firewalls_chart.html.
- Managed virtual private networking chart, www.isp-planet.com/technology/mssp/services_chart.html.
- Managed intrusion detection and security monitoring, www.isp-planet.com/technology/mssp/monitoring_chart.html.
- Managed antivirus and managed content filtering and URL blocking, www.isp-planet.com/technology/mssp/mssp_survey2.html.
- Managed vulnerability assessment and emergency response and forensics, www.isp-planet.com/technology/mssp/mssp_survey3.html.

Exercise 7.01 introduces you to WinDump. This tool is similar to the Linux tool TCPDump. It is a simple packet-capture program that can be used to help demonstrate how IDS systems work. All IDS systems must first capture packets so that the traffic can be analyzed.

Configuring & Implementing…

Installing WinDUMP for Packet Capture and ANALYSIS

1. Go to www.winpcap.org/windump/install/
2. At the top of the page you will see a link for WinPcap. This program will need to be installed as it will allow the capture of low level packets.
3. Next, download and install the WinDump program from the link indicated on the same Web page.
4. You'll now need to open a command prompt by clicking Start, Run and entering cmd in the Open Dialog box.
5. With a command prompt open, you can now start the program by typing WinDump from the command line. By default, it will use the first Ethernet adaptor found. You can display the help screen by typing **windump –h**. The example below specifies the second adaptor.

Continued

```
C:\>windump -i 2
```

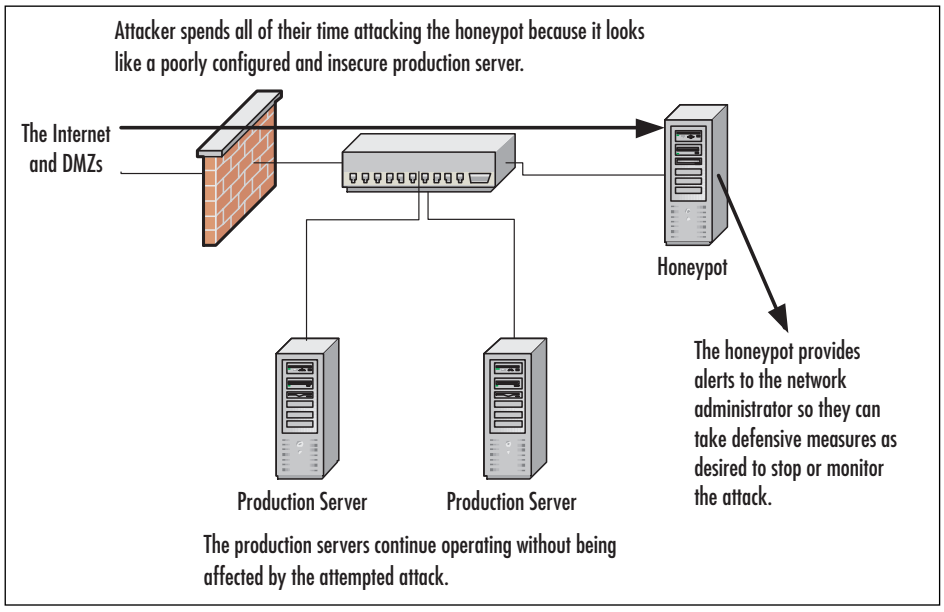
6. You should now see the program running. If there is little traffic on your network, you can open a second command prompt and ping a host such as www.yahoo.com. The results should be seen in the screen you have open that is running WinDump as seen below.

```
windump: listening on \Device\eth0_
14:07:02.563213 IP earth.137 > 192.168.123.181.137: UDP, length 50
14:07:04.061618 IP earth.137 > 192.168.123.181.137: UDP, length 50
14:07:05.562375 IP earth.137 > 192.168.123.181.137: UDP, length 50
```

Honeypots and Honeynets

A *honeypot* is a computer system that is deliberately exposed to public access—usually on the Internet—for the express purpose of attracting and distracting attackers. Likewise, a *honeynet* is a network set up for the same purpose, where attackers not only find vulnerable services or servers but also find vulnerable routers, firewalls, and other network boundary devices, security applications, and so forth. In other words, these are the technical equivalent of the familiar police "sting" operation. Although the strategy involved in luring hackers to spend time investigating attractive network devices or servers can cause its own problems, finding ways to lure intruders into a system or network improves the odds of being able to identify those intruders and pursue them more effectively. Figure 7.14 shows a graphical representation of the honeypot concept in action.

Figure 7.14 A Honeypot in Use to Keep Attackers from Affecting Critical Production Servers



Notes from the Underground…

Walking the Line Between Opportunity and Entrapment

Most law enforcement officers are aware of the fine line they must walk when setting up a "sting"—an operation in which police officers pretend to be victims or participants in crime, with the goal of getting criminal suspects to commit an illegal act in their presence. Most states have laws that prohibit entrapment; that is, law enforcement officers are not allowed to *cause* a person to commit a crime and then arrest him or her for doing it. Entrapment is a defense to prosecution; if the accused person can show at trial that he or she was entrapped, the result must be an acquittal.

Courts have traditionally held, however, that providing a *mere opportunity* for a criminal to commit a crime does not constitute entrapment. To entrap involves using persuasion, duress, or other undue pressure to force someone to commit a crime that the person would not otherwise have committed. Under this holding, setting up a honeypot or honeynet would be like the (perfectly legitimate) police tactic of placing an abandoned automobile by the side of the road and watching it to see if anyone attempts to burglarize, vandalize, or steal it. It should also be noted that entrapment only applies to the actions of law enforcement or government personnel. A civilian cannot entrap, regardless of how much pressure is exerted on the target to commit the crime. (However, a civilian could be subject to other charges, such as criminal solicitation or criminal conspiracy, for causing someone else to commit a crime.)

The following characteristics are typical of honeypots or honeynets:

- Systems or devices used as lures are set up with only "out of the box" default installations, so that they are deliberately made subject to all known vulnerabilities, exploits, and attacks.
- The systems or devices used as lures do not include sensitive information (e.g., passwords, data, applications, or services an organization depends on or must absolutely protect), so these lures can be compromised, or even destroyed, without causing damage, loss, or harm to the organization that presents them to be attacked.
- Systems or devices used as lures often also contain deliberately tantalizing objects or resources, such as files named *password.db,* folders named *Top Secret*, and so forth—often consisting only of encrypted garbage data or log files of no real significance or value—to attract and hold an attacker's interest long enough to give a backtrace a chance of identifying the attack's point of origin.
- Systems or devices used as lures also include or are monitored by passive applications that can detect and report on attacks or intrusions as soon as they start, so the process of backtracing and identification can begin as soon as possible.

Although this technique can help identify the unwary or unsophisticated attacker, it also runs the risk of attracting additional attention from savvier attackers. Honeypots or honeynets, once identified, are often publicized on hacker message boards or mailing lists and thus become *more* subject to attacks and hacker activity than they otherwise might be. Likewise, if the organization that sets up a honeypot or honeynet is itself identified, its production systems and networks may also be subjected to more attacks than might otherwise occur.

NOTE

A *honeypot* is a computer system that is deliberately exposed to public access—usually on the Internet—for the express purpose of attracting and distracting attackers. Likewise, a *honeynet* is a network set up for the same purpose, where attackers not only find vulnerable services or servers, but also find vulnerable routers, firewalls, and other network boundary devices, security applications, and so forth. You

The honeypot technique is best reserved for use when a company or organization employs full-time Information Technology (IT) security professionals who can monitor and deal with these lures on a regular basis, or when law enforcement operations seek to target specific suspects in a "virtual sting" operation. In such situations, the risks are sure to be well understood, and proper security precautions, processes, and procedures are far more likely to already be in place (and properly practiced). Nevertheless, for organizations that seek to identify and pursue attackers more proactively, honeypots and honeynets can provide valuable tools to aid in such activities.

Although numerous quality resources on honeypots and honeynets are available (try searching on either term at www.searchsecurity.techtarget.com), the following resources are particularly valuable for people seeking additional information on the topic. John McMullen's article "Enhance Intrusion Detection with a Honeypot" at www.techrepublic.com/article_guest.jhtml?id=r00220010412mul01.htm&fromtm=e036 sheds additional light on this topic. The Honeynet Project at http://www.honeynet.org is probably the best overall resource on the topic online; it not only provides copious information on the project's work to define and document standard honeypots and honeynets, it also does a great job of exploring hacker mindsets, motivations, tools, and attack techniques.

Judging False Positives and Negatives

As mentioned earlier, understanding the state of an IDS is very important. To be an effective tool, an IDS must be configured properly. A false positive is a triggered event that did not actually occur, which may be as innocuous as the download of a signature database (downloading of an IDS signature database may trigger every alarm in the book) or some unusual traffic generated by a networked game. False positives have a significant impact on the effectiveness of an IDS sensor. If there are a reasonable number of false positives being detected, the perceived urgency of an alert may be diminished by the fact that there are numerous events being triggered on a daily basis that turn into wild goose chases. In the end, all the power of IDS is ultimately controlled by a single judgment call on whether or not to take action.

More dangerous, however, is the possibility for a false negative, which is the failure to be alerted to an actual event. This would occur in a failure of one of the key functional units of a NIDS. False negatives can occur because of misconfigurations when an attacker modifies the attack payload in order to subvert the detection engine.

WARNING

A false positive is defined as a positive detection result that is false or untrue. This can be dangerous because you may spend wasted time trying to put together the facts of the case and look for a weakness in your system. A false negative, on the other hand, is a negative detection event that is actually positive or true. False negatives are the worst of the four states that can occur in an IDS. A false negative gives you the feeling that everything is OK, all the while an attacker has comprised your systems and is helping themselves to your sensitive and valuable data.

Incident Response

The first thing that must be done after receiving notification of an attack is to respond to the attack. In some cases the administrator may want to allow the attack to continue for a short period of time so that they can collect further data and other evidence about the attack, its origin, and its methods. After terminating the attack, or upon discovering the evidence of the attack, they must take all available steps to ensure that the chain of evidence will not be lost. They must save and export log and audit files, close open ports that have been exploited, and secure services that should not have been running in the first place. In short, take every available step to ensure that the same type of attack will not occur again some time in the future.

Summary

In today's networking world, networks no longer have to be designed the same way. There are many options available as to how to physically and logically design a network. All of these options can be used to increase the security of the internal network by keeping untrusted and unauthorized users out. The usage of DMZs to segment Web and e-mail traffic to a protected zone between external and internal firewalls, helps prevent attacks that may deface the Web server from having any effect on the critical database servers. Just the same, an attack on your Web server will have little lasting damage.

A NAT device can be used to hide the private intranet from the public Internet. NAT devices work by translating all private IP addresses into one or more public IP addresses, therefore making it look as if all traffic from the internal network is coming from one computer (or a small group of computers). The NAT device maintains a routing table of all connection requests, and therefore is able to ensure that all returning packets get directed to the correct originating host. Extranets can be established using VPN tunnels to provide secure access to intranet resources from different geographic locations. VPNs are also used to allow remote network users to securely connect back to the corporate network.

IDSes are used to identify and respond to attacks on the network. Several types of IDSes exist, each with its own unique pros and cons. Which type you choose depends on your needs, and ultimately on your budget. An IPS is a newer type of IDS that can quickly respond to perceived attacks. Honeypots are advanced IDSes that can intelligently respond to attacks, actually enticing the attacker to select them over other real targets on the network. Honeypots can be used to distract attackers from real servers and keep them occupied while you collect information on the attack and the source of the attack.

After an attack has occurred, the most important thing to do is to collect all of the evidence of the attack and its methods. You will also want to take steps to ensure that the same type of attack cannot be successfully performed on the network in the future.

Solutions Fast Track

Security Topologies

- ☑ A DMZ is a network segment where systems that are accessible to the public Internet are housed and which offers some basic levels of protection against attacks.
- ☑ The creation of DMZ segments is usually done by placing systems between two firewall devices that have different rule sets. This allows systems on the Internet to connect to the offered services on the DMZ systems but not to the computers on the internal segments of the organization (often called the protected network).
- ☑ A private internal network is called the intranet, as opposed to the Internet (which is the large publicly accessible network). It is expected that all traffic on an intranet will be secure from outside attack or compromise.

- ☑ An extranet is a special topology that is implemented in certain cases where there is a need to allow access to some of the internal network data and resources by users outside of the internal network.
- ☑ Using special features found in newer, more expensive switches and special software in the switch, you can physically split one switch into two, thus creating two network segments that are completely separate from one another and creating a VLAN.
- ☑ NAT is a feature of many firewalls, proxies, and routing-capable systems. NAT has several benefits, one of which is its ability to hide the IP addresses and network design of the internal network. The ability to hide the internal network from the Internet reduces the risk of intruders gleaning information about the network and exploiting that information to gain access. If an intruder does not know the structure of a network, the network layout, the names and IP address of systems, and so on, it is very difficult to gain access to that network.
- ☑ Tunneling is used to create a virtual point-to-point connection between you and your destination using an untrusted public network as the medium. In most cases, this would be the Internet. When you establish a secure tunnel, commonly called a VPN, you are creating a safe connection between two points that cannot be examined by outsiders. All packets are encrypted and carry information that ensure they are tamperproof and thus can withstand common IP attacks, such as the MITM and packet replay. When a VPN is created, you can be reasonably secure that the traffic is private and safe from prying eyes.

Intrusion Detection

- ☑ An IDS is a specialized tool that knows how to read and interpret the contents of log files from routers, firewalls, servers, and other network devices. Furthermore, an IDS often stores a database of known attack signatures and can compare patterns of activity, traffic, or behavior it sees in the logs it is monitoring against those signatures to recognize when a close match between a signature and current or recent behavior occurs. At that point, the IDS can issue alarms or alerts, take various kinds of automatic action ranging from shutting down Internet links or specific servers to launching backtraces, and make other active attempts to identify attackers and actively collect evidence of their nefarious activities.
- ☑ IDSes that monitor network backbones and look for attack signatures are called network-based IDSes, whereas those that operate on hosts defend and monitor the operating and file systems for signs of intrusion and are called host-based IDSes. Some IDSes monitor only specific applications and are called application-based IDSes. (This type of treatment is usually reserved for important applications such as database management systems, content management systems, accounting systems, and so forth.)
- ☑ IDSes may also be distinguished by their differing approaches to event analysis. Some IDSes primarily use a technique called signature detection. This resembles the way many antivirus programs use virus signatures to recognize and block infected files, programs, or active Web content from entering a computer system, except that it uses a database of traffic or activity patterns related to known attacks, called attack signatures. Signature detection is the most

widely used approach in commercial IDS technology today. Another approach is called anomaly detection. It uses rules or predefined concepts about "normal" and "abnormal" system activity (called heuristics) to distinguish anomalies from normal system behavior and to monitor, report on, or block anomalies as they occur.

☑ A honeypot is a computer system that is deliberately exposed to public access—usually on the Internet—for the express purpose of attracting and distracting attackers. Likewise, a honeynet is a network set up for the same purpose, where attackers find vulnerable services or servers and also find vulnerable routers, firewalls, and other network boundary devices, security applications, and so forth.

Frequently Asked Questions

The following Frequently Asked Questions, answered by the authors of this book, are designed to both measure your understanding of the concepts presented in this chapter and to assist you with real-life implementation of these concepts. To have your questions about this chapter answered by the author, browse to **www.syngress.com/solutions** and click on the **"Ask the Author"** form.

Q: Why do I need to create a DMZ for my Web and e-mail servers? Can't I just put all of my computers behind my firewall on my intranet?

A: You can, but by doing so you open yourself up to all sorts of attacks that you would otherwise be protected from if you allow outside users to access any of those resources. You need a DMZ if you want to make certain resources available to outside users over the Internet (for example, if you want to host a Web server). By placing certain computers, such as Web servers and front-end e-mail servers, on a DMZ, you can keep these often abused ports controlled on your internal firewall (by controlling access by IP address), thus lessoning the chance of a successful attack on your intranet.

Q: What advantage does a honeypot offer me over a traditional IDS system?

A: A honeypot is a very intelligent IDS that not only monitors an attacker, but also interacts with attackers, keeping them interested in the honeypot and away from the real production servers on your network. While the attacker is distracted and examining the non-critical data they find in the honeypot, you have more time to track the attacker's identity.

Q: What is the difference between an Internet, intranet, and extranet? Aren't they all terms for the same thing?

A: The Internet is a network of networks that are connected together and is the biggest public network in existence, which grew out of the ARPANet project. An intranet is a private internal network available to users within the organization, whereas an extranet is a special topology that is implemented in certain cases where you have a need to allow access to some of your internal network data and resources by users outside of your internal network.

Q: What type of IDS should I choose?

A: The type of IDS you choose to employ on your network will depend on what type of network you have and what types of applications you are running. Host-based IDSes can effectively monitor one specific computer, but not the entire network. Network-based IDSes can monitor the entire network from a high-level view, but may miss some type of attacks. Application-based IDSes are specific to one application, such as a database application, and will monitor attacks only on that application.

Q: Why would I want to use a VLAN?

A: VLANs can be used to segment network traffic into different broadcast domains. This adds another layer of security for your network by keeping certain traffic segmented from the rest of your network traffic—all inside of your firewall.