

application gateway, a circuit-level gateway does not permit an end-to-end TCP connection; rather, the gateway sets up two TCP connections, one between itself and a TCP user on an inner host and one between itself and a TCP user on an outside host. Once the two connections are established, the gateway typically relays TCP segments from one connection to the other without examining the contents. The security function consists of determining which connections will be allowed.

A typical use of circuit-level gateways is a situation in which the system administrator trusts the internal users. The gateway can be configured to support application-level or proxy service on inbound connections and circuit-level functions for outbound connections. In this configuration, the gateway can incur the processing overhead of examining incoming application data for forbidden functions but does not incur that overhead on outgoing data.

An example of a circuit-level gateway implementation is the SOCKS package [KOBL92]; version 5 of SOCKS is specified in RFC 1928. The SOCKS protocol provides a framework for client-server applications in both the TCP and UDP domains. It is designed to provide convenient and secure access to a network-level firewall. The protocol occupies a thin layer between the application and either TCP or UDP but does not provide network-level routing services, such as forwarding of ICMP messages.

SOCKS consists of the following components:

- The SOCKS server, which often runs on a UNIX-based firewall. SOCKS is also implemented on Windows systems.
- The SOCKS client library, which runs on internal hosts protected by the firewall.
- SOCKS-ified versions of several standard client programs such as FTP and TELNET. The implementation of the SOCKS protocol typically involves either the recompilation or relinking of TCP-based client applications or the use of alternate dynamically loaded libraries, to use the appropriate encapsulation routines in the SOCKS library.

When a TCP-based client wishes to establish a connection to an object that is reachable only via a firewall (such determination is left up to the implementation), it must open a TCP connection to the appropriate SOCKS port on the SOCKS server system. The SOCKS service is located on TCP port 1080. If the connection request succeeds, the client enters a negotiation for the authentication method to be used, authenticates with the chosen method, and then sends a relay request. The SOCKS server evaluates the request and either establishes the appropriate connection or denies it. UDP exchanges are handled in a similar fashion. In essence, a TCP connection is opened to authenticate a user to send and receive UDP segments, and the UDP segments are forwarded as long as the TCP connection is open.

## 12.4 FIREWALL BASING

It is common to base a firewall on a stand-alone machine running a common operating system, such as UNIX or Linux. Firewall functionality can also be implemented as a software module in a router or LAN switch. In this section, we look at some additional firewall basing considerations.

## Bastion Host

A bastion host is a system identified by the firewall administrator as a critical strong point in the network's security. Typically, the bastion host serves as a platform for an application-level or circuit-level gateway. Common characteristics of a bastion host are as follows:

- The bastion host hardware platform executes a secure version of its operating system, making it a hardened system.
- Only the services that the network administrator considers essential are installed on the bastion host. These could include proxy applications for DNS, FTP, HTTP, and SMTP.
- The bastion host may require additional authentication before a user is allowed access to the proxy services. In addition, each proxy service may require its own authentication before granting user access.
- Each proxy is configured to support only a subset of the standard application's command set.
- Each proxy is configured to allow access only to specific host systems. This means that the limited command/feature set may be applied only to a subset of systems on the protected network.
- Each proxy maintains detailed audit information by logging all traffic, each connection, and the duration of each connection. The audit log is an essential tool for discovering and terminating intruder attacks.
- Each proxy module is a very small software package specifically designed for network security. Because of its relative simplicity, it is easier to check such modules for security flaws. For example, a typical UNIX mail application may contain over 20,000 lines of code, while a mail proxy may contain fewer than 1000.
- Each proxy is independent of other proxies on the bastion host. If there is a problem with the operation of any proxy, or if a future vulnerability is discovered, it can be uninstalled without affecting the operation of the other proxy applications. Also, if the user population requires support for a new service, the network administrator can easily install the required proxy on the bastion host.
- A proxy generally performs no disk access other than to read its initial configuration file. Hence, the portions of the file system containing executable code can be made read only. This makes it difficult for an intruder to install Trojan horse sniffers or other dangerous files on the bastion host.
- Each proxy runs as a nonprivileged user in a private and secured directory on the bastion host.

## Host-Based Firewalls

A host-based firewall is a software module used to secure an individual host. Such modules are available in many operating systems or can be provided as an add-on package. Like conventional stand-alone firewalls, host-resident firewalls filter

and restrict the flow of packets. A common location for such firewalls is a server. There are several advantages to the use of a server-based or workstation-based firewall:

- Filtering rules can be tailored to the host environment. Specific corporate security policies for servers can be implemented, with different filters for servers used for different application.
- Protection is provided independent of topology. Thus both internal and external attacks must pass through the firewall.
- Used in conjunction with stand-alone firewalls, the host-based firewall provides an additional layer of protection. A new type of server can be added to the network, with its own firewall, without the necessity of altering the network firewall configuration.

### Personal Firewall

A personal firewall controls the traffic between a personal computer or workstation on one side and the Internet or enterprise network on the other side. Personal firewall functionality can be used in the home environment and on corporate intranets. Typically, the personal firewall is a software module on the personal computer. In a home environment with multiple computers connected to the Internet, firewall functionality can also be housed in a router that connects all of the home computers to a DSL, cable modem, or other Internet interface.

Personal firewalls are typically much less complex than either server-based firewalls or stand-alone firewalls. The primary role of the personal firewall is to deny unauthorized remote access to the computer. The firewall can also monitor outgoing activity in an attempt to detect and block worms and other malware.

Personal firewall capabilities are provided by the netfilter package on Linux systems, or the pf package on BSD and Mac OS X systems. These packages may be configured on the command-line, or with a GUI front-end. When such a personal firewall is enabled, all inbound connections are usually denied except for those the user explicitly permits. Outbound connections are usually allowed. The list of inbound services that can be selectively re-enabled, with their port numbers, may include the following common services:

- Personal file sharing (548, 427)
- Windows sharing (139)
- Personal Web sharing (80, 427)
- Remote login—SSH (22)
- FTP access (20–21, 1024–65535 from 20–21)
- Printer sharing (631, 515)
- iChat Rendezvous (5297, 5298)
- iTunes Music Sharing (3869)
- CVS (2401)
- Gnutella/Limewire (6346)

- ICQ (4000)
- IRC (194)
- MSN Messenger (6891–6900)
- Network Time (123)
- Retrospect (497)
- SMB (without netbios; 445)
- Timbuktu (407)
- VNC (5900–5902)
- WebSTAR Admin (1080, 1443)

When FTP access is enabled, ports 20 and 21 on the local machine are opened for FTP; if others connect this computer from ports 20 or 21, the ports 1024 through 65535 are open.

For increased protection, advanced firewall features may be configured. For example, stealth mode hides the system on the Internet by dropping unsolicited communication packets, making it appear as though the system is not present. UDP packets can be blocked, restricting network traffic to TCP packets only for open ports. The firewall also supports logging, an important tool for checking on unwanted activity. Other types of personal firewall allow the user to specify that only selected applications, or applications signed by a valid certificate authority, may provide services accessed from the network.

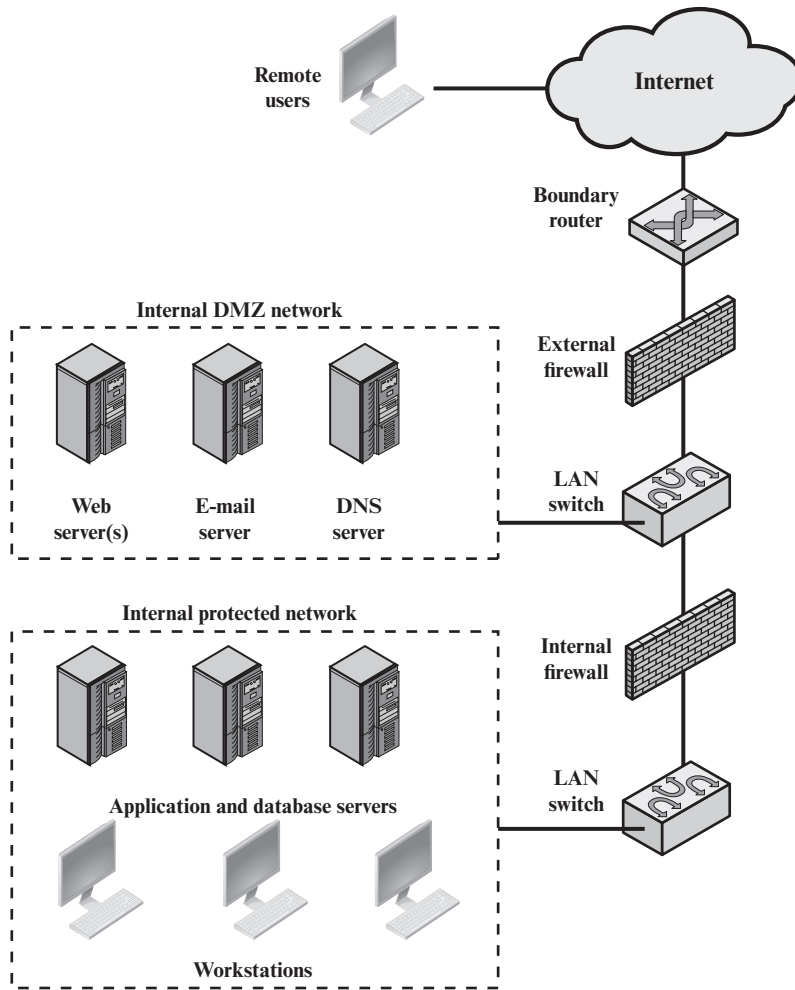
## 12.5 FIREWALL LOCATION AND CONFIGURATIONS

As Figure 12.1a indicates, a firewall is positioned to provide a protective barrier between an external, potentially untrusted source of traffic and an internal network. With that general principle in mind, a security administrator must decide on the location and on the number of firewalls needed. In this section, we look at some common options.

### DMZ Networks

Figure 12.2 suggests the most common distinction, that between an internal and an external firewall. An external firewall is placed at the edge of a local or enterprise network, just inside the boundary router that connects to the Internet or some wide area network (WAN). One or more internal firewalls protect the bulk of the enterprise network. Between these two types of firewalls are one or more networked devices in a region referred to as a DMZ (demilitarized zone) network. Systems that are externally accessible but need some protections are usually located on DMZ networks. Typically, the systems in the DMZ require or foster external connectivity, such as a corporate Web site, an e-mail server, or a DNS (domain name system) server.

The external firewall provides a measure of access control and protection for the DMZ systems consistent with their need for external connectivity. The external



**Figure 12.2** Example Firewall Configuration

firewall also provides a basic level of protection for the remainder of the enterprise network. In this type of configuration, internal firewalls serve three purposes:

1. The internal firewall adds more stringent filtering capability, compared to the external firewall, in order to protect enterprise servers and workstations from external attack.
2. The internal firewall provides two-way protection with respect to the DMZ. First, the internal firewall protects the remainder of the network from attacks launched from DMZ systems. Such attacks might originate from worms, root-kits, bots, or other malware lodged in a DMZ system. Second, an internal firewall can protect the DMZ systems from attack from the internal protected network.

3. Multiple internal firewalls can be used to protect portions of the internal network from each other. For example, firewalls can be configured so that internal servers are protected from internal workstations and vice versa. A common practice is to place the DMZ on a different network interface on the external firewall from that used to access the internal networks.

## Virtual Private Networks

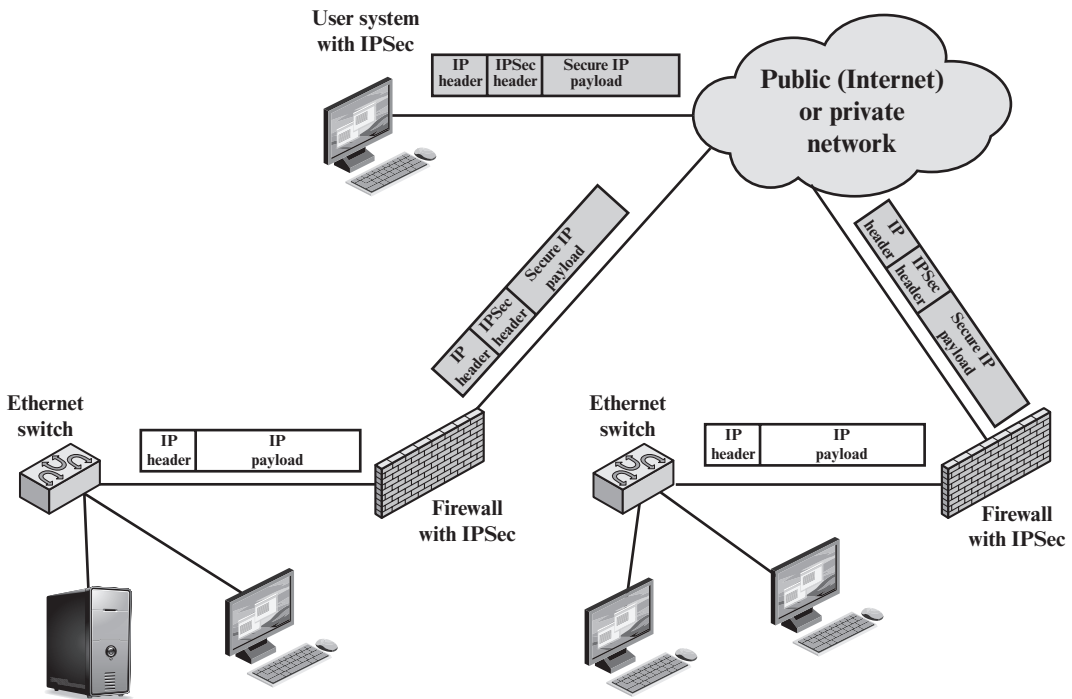
In today's distributed computing environment, the **virtual private network** (VPN) offers an attractive solution to network managers. In essence, a VPN consists of a set of computers that interconnect by means of a relatively unsecure network and that make use of encryption and special protocols to provide security. At each corporate site, workstations, servers, and databases are linked by one or more local area networks (LANs). The Internet or some other public network can be used to interconnect sites, providing a cost savings over the use of a private network and offloading the wide area network management task to the public network provider. That same public network provides an access path for telecommuters and other mobile employees to log on to corporate systems from remote sites.

But the manager faces a fundamental requirement: security. Use of a public network exposes corporate traffic to eavesdropping and provides an entry point for unauthorized users. To counter this problem, a VPN is needed. In essence, a VPN uses encryption and authentication in the lower protocol layers to provide a secure connection through an otherwise insecure network, typically the Internet. VPNs are generally cheaper than real private networks using private lines but rely on having the same encryption and authentication system at both ends. The encryption may be performed by firewall software or possibly by routers. The most common protocol mechanism used for this purpose is at the IP level and is known as IPsec.

Figure 12.3 (Compare Figure 9.1) is a typical scenario of IPsec usage.<sup>1</sup> An organization maintains LANs at dispersed locations. Nonsecure IP traffic is conducted on each LAN. For traffic off site, through some sort of private or public WAN, IPsec protocols are used. These protocols operate in networking devices, such as a router or firewall, that connect each LAN to the outside world. The IPsec networking device will typically encrypt and compress all traffic going into the WAN and decrypt and uncompress traffic coming from the WAN; authentication may also be provided. These operations are transparent to workstations and servers on the LAN. Secure transmission is also possible with individual users who dial into the WAN. Such user workstations must implement the IPsec protocols to provide security. They must also implement high levels of host security, as they are directly connected to the wider Internet. This makes them an attractive target for attackers attempting to access the corporate network.

A logical means of implementing an IPsec is in a firewall, as shown in Figure 12.3. If IPsec is implemented in a separate box behind (internal to) the firewall, then VPN traffic passing through the firewall in both directions is encrypted. In this case, the firewall is unable to perform its filtering function or other security

<sup>1</sup>Details of IPsec are provided in Chapter 9. For this discussion, all that we need to know is that IPsec adds one or more additional headers to the IP packet to support encryption and authentication functions.



**Figure 12.3** A VPN Security Scenario

functions, such as access control, logging, or scanning for viruses. IPSec could be implemented in the boundary router, outside the firewall. However, this device is likely to be less secure than the firewall and thus less desirable as an IPSec platform.

### Distributed Firewalls

A distributed firewall configuration involves stand-alone firewall devices plus host-based firewalls working together under a central administrative control. Figure 12.4 suggests a distributed firewall configuration. Administrators can configure host-resident firewalls on hundreds of servers and workstations as well as configure personal firewalls on local and remote user systems. Tools let the network administrator set policies and monitor security across the entire network. These firewalls protect against internal attacks and provide protection tailored to specific machines and applications. Stand-alone firewalls provide global protection, including internal firewalls and an external firewall, as discussed previously.

With distributed firewalls, it may make sense to establish both an internal and an external DMZ. Web servers that need less protection because they have less critical information on them could be placed in an external DMZ, outside the external firewall. What protection is needed is provided by host-based firewalls on these servers.

An important aspect of a distributed firewall configuration is security monitoring. Such monitoring typically includes log aggregation and analysis, firewall statistics, and fine-grained remote monitoring of individual hosts if needed.

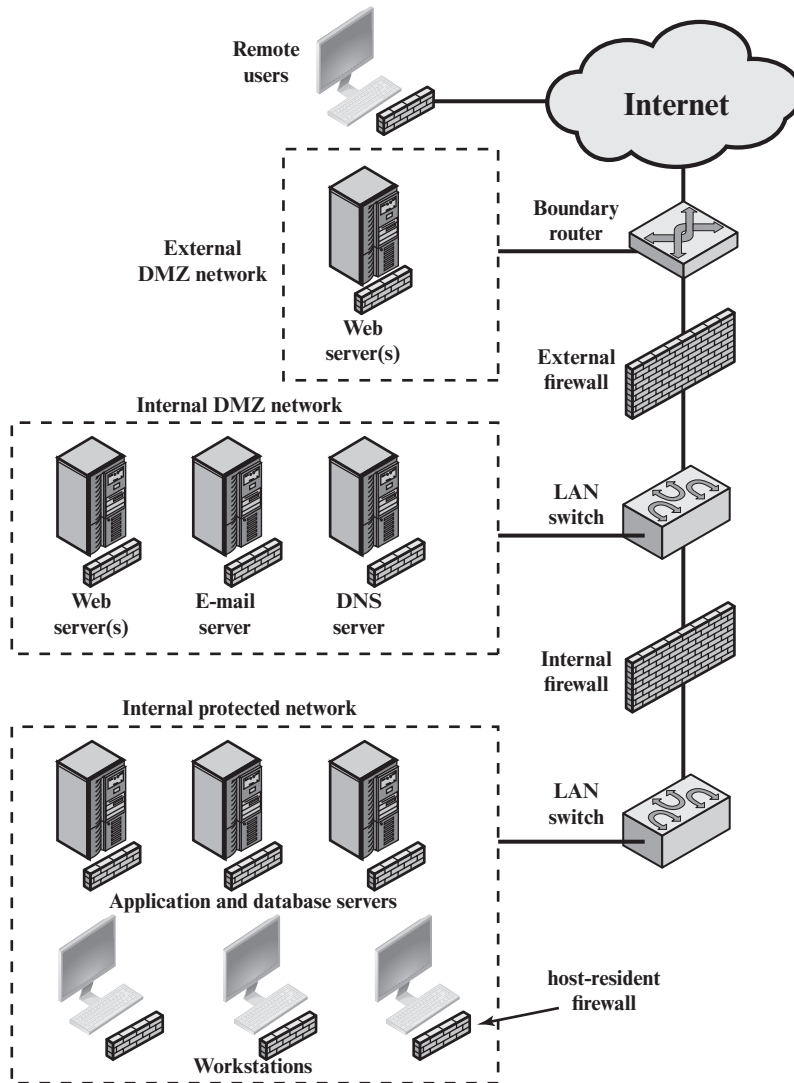


Figure 12.4 Example Distributed Firewall Configuration

## Summary of Firewall Locations and Topologies

We can now summarize the discussion from Sections 12.4 and 12.5 to define a spectrum of firewall locations and topologies. The following alternatives can be identified:

- **Host-resident firewall:** This category includes personal firewall software and firewall software on servers. Such firewalls can be used alone or as part of an in-depth firewall deployment.



- **Screening router:** A single router between internal and external networks with stateless or full packet filtering. This arrangement is typical for small office/home office (SOHO) applications.
- **Single bastion inline:** A single firewall device between an internal and external router (e.g., Figure 12.1a). The firewall may implement stateful filters and/or application proxies. This is the typical firewall appliance configuration for small- to medium-sized organizations.
- **Single bastion T:** Similar to single bastion inline but has a third network interface on bastion to a DMZ where externally visible servers are placed. Again, this is a common appliance configuration for medium to large organizations.
- **Double bastion inline:** Figure 12.2 illustrates this configuration, where the DMZ is sandwiched between bastion firewalls. This configuration is common for large businesses and government organizations.
- **Double bastion T:** The DMZ is on a separate network interface on the bastion firewall. This configuration is also common for large businesses and government organizations and may be required. For example, this configuration is required for Australian government use (Australian Government Information Technology Security Manual—ACSI33).
- **Distributed firewall configuration:** Illustrated in Figure 12.4. This configuration is used by some large businesses and government organizations.

## 12.6 KEY TERMS, REVIEW QUESTIONS, AND PROBLEMS

### Key Terms

application-level gateway bastion host circuit-level gateway distributed firewalls DMZ	firewall host-based firewall IP address spoofing IP security (IPSec) packet filtering firewall	personal firewall proxy stateful inspection firewall tiny fragment attack virtual private network (VPN)
--	--	---

### Review Questions

- 12.1 List three design goals for a firewall.
- 12.2 List four techniques used by firewalls to control access and enforce a security policy.
- 12.3 When does a packet filtering firewall resort to default actions? List these default policies.
- 12.4 What are some weaknesses of a packet filtering firewall?
- 12.5 Explain three attacks that can be made on packet filtering firewalls. What measures can be taken to counter these attacks?
- 12.6 What is an application-level gateway?
- 12.7 What is a circuit-level gateway?
- 12.8 What are the differences among the firewalls of Figure 12.1?
- 12.9 What are the common characteristics of a bastion host?

- 12.10** Why is it useful to have host-based firewalls?
- 12.11** What is a virtual private network? How does it ensure a secure connection?
- 12.12** Describe the spectrum of firewall locations and topologies.

## Problems

- 12.1** As was mentioned in Section 12.3, one approach to defeating the tiny fragment attack is to enforce a minimum length of the transport header that must be contained in the first fragment of an IP packet. If the first fragment is rejected, all subsequent fragments can be rejected. However, the nature of IP is such that fragments may arrive out of order. Thus, an intermediate fragment may pass through the filter before the initial fragment is rejected. How can this situation be handled?
- 12.2** In an IPv4 packet, the size of the payload in the first fragment, in octets, is equal to  $\text{Total Length} - (4 \times \text{IHL})$ . If this value is less than the required minimum (8 octets for TCP), then this fragment and the entire packet are rejected. Suggest an alternative method of achieving the same result using only the Fragment Offset field.
- 12.3** RFC 791, the IPv4 protocol specification, describes a reassembly algorithm that results in new fragments overwriting any overlapped portions of previously received fragments. Given such a reassembly implementation, an attacker could construct a series of packets in which the lowest (zero-offset) fragment would contain innocuous data (and thereby be passed by administrative packet filters), and in which some subsequent packet having a non-zero offset would overlap TCP header information (destination port, for instance) and cause it to be modified. The second packet would be passed through most filter implementations because it does not have a zero fragment offset. Suggest a method that could be used by a packet filter to counter this attack.
- 12.4** Table 12.3 shows a sample of a packet filter firewall ruleset for an imaginary network of IP address that range from 192.168.1.0 to 192.168.1.254. Describe the effect of each rule.
- 12.5** SMTP (Simple Mail Transfer Protocol) is the standard protocol for transferring mail between hosts over TCP. A TCP connection is set up between a user agent and a server program. The server listens on TCP port 25 for incoming connection requests. The user end of the connection is on a TCP port number above 1023. Suppose you wish to build a packet filter ruleset allowing inbound and outbound SMTP traffic. You generate the following ruleset:

**Table 12.3** Sample Packet Filter Firewall Ruleset

	Source Address	Source Port	Dest Address	Dest Port	Action
<b>1</b>	Any	Any	192.168.1.0	> 1023	Allow
<b>2</b>	192.168.1.1	Any	Any	Any	Deny
<b>3</b>	Any	Any	192.168.1.1	Any	Deny
<b>4</b>	192.168.1.0	Any	Any	Any	Allow
<b>5</b>	Any	Any	192.168.1.2	SMTP	Allow
<b>6</b>	Any	Any	192.168.1.3	HTTP	Allow
<b>7</b>	Any	Any	Any	Any	Deny