National Institute of
Standards and Technology
U.S. Department of Commerce

Special Publication 800-41
Revision 1 (Draft)

Guidelines on Firewalls and Firewall Policy (Draft)

Recommendations of the National Institute of Standards and Technology

Karen Scarfone
Paul Hoffman

NIST Special Publication 800-41
Revision 1 (Draft)

Guidelines on Firewalls and Firewall Policy (Draft)

Recommendations of the National Institute of Standards and Technology

Karen Scarfone
Paul Hoffman

C O M P U T E R S E C U R I T Y

Computer Security Division
Information Technology Laboratory
National Institute of Standards and Technology
Gaithersburg, MD 20899-8930

July 2008

U.S. Department of Commerce

Carlos M. Gutierrez, Secretary

National Institute of Standards and Technology

James M. Turner, Deputy Director

Reports on Computer Systems Technology

The Information Technology Laboratory (ITL) at the National Institute of Standards and Technology (NIST) promotes the U.S. economy and public welfare by providing technical leadership for the nation's measurement and standards infrastructure. ITL develops tests, test methods, reference data, proof of concept implementations, and technical analysis to advance the development and productive use of information technology. ITL's responsibilities include the development of technical, physical, administrative, and management standards and guidelines for the cost-effective security and privacy of sensitive unclassified information in Federal computer systems. This Special Publication 800-series reports on ITL's research, guidance, and outreach efforts in computer security and its collaborative activities with industry, government, and academic organizations.

National Institute of Standards and Technology Special Publication 800-41 Revision 1 (Draft)
Natl. Inst. Stand. Technol. Spec. Publ. 800-41 rev1, 43 pages (Jul. 2008)

Acknowledgements

The authors, Karen Scarfone of the National Institute of Standards and Technology (NIST) and Paul Hoffman of the Virtual Private Network Consortium (VPNC), wish to thank their colleagues who reviewed drafts of this document and contributed to its technical content. The authors would like to acknowledge Tim Grance of NIST, and Matthew Goche, David Klug, Logan Lodge, John Pearce, Noel Richards, Anne Roudabush, and Steven Sharma of Booz Allen Hamilton, for their keen and insightful assistance throughout the development of the document. Special thanks go to Brahim Asfahani of Booz Allen Hamilton for his contributions to early drafts of the document.

The authors also wish to express their thanks to the individuals and organizations that contributed to the original version of the publication, including John Wack of NIST and Ken Cutler and Jamie Pole of the MIS Training Institute, who authored the original version, and other contributors and reviewers—particularly Peter Batista and Wayne Bavry (U.S. Treasury); Harriet Feldman (Integrated Computer Engineering, Inc.); Rex Sanders (U.S. Geological Survey); and Timothy Grance, D. Richard Kuhn, Peter Mell, Gale Richter, and Murugiah Souppaya (NIST).

Additional acknowledgements will be added to the final version of the publication.

Table of Contents

List of Appendices

List of Figures

List of Tables

Executive Summary

Firewalls are devices or programs that control the flow of network traffic between networks or hosts that employ differing security postures. At one time, most firewalls were deployed at network perimeters. This provided some measure of protection for internal hosts, but it could not recognize all instances and forms of attack, and attacks sent from one internal host to another often do not pass through network firewalls. Because of these and other factors, network designers now often include firewall functionality at places other than the network perimeter to provide an additional layer of security, as well as to protect mobile devices that are placed directly onto external networks. Also, threats have gradually moved from lower layers of network traffic to the application layer, reducing the effectiveness of firewalls that focus on lower layers.

There are several types of firewalls, each with varying capabilities to analyze network traffic and allow or block specific instances by comparing traffic characteristics to existing policies. Understanding the capabilities of each type of firewall, and designing firewall policies and acquiring firewall technologies that effectively address an organization's needs, are critical to achieving protection for network traffic flows. This document provides an overview of several types of firewall technologies, and discusses their security capabilities and relative advantages and disadvantages in detail. It also provides several examples of where firewalls can be placed within networks, and the implications of deploying firewalls in particular locations. The document also makes recommendations for establishing firewall policies and for selecting, configuring, testing, deploying, and managing firewall solutions.

To improve the effectiveness and security of their firewalls, organizations should implement the following recommendations:

Create a firewall policy that specifies how firewalls should handle network traffic.

A firewall policy defines how an organization's firewalls should handle network traffic for specific IP addresses and address ranges, protocols, applications, and content types based on the organization's information security policies. Organizations should conduct risk analysis to develop a list of the types of traffic needed by the organization and how they must be secured—including which types of traffic can traverse a firewall under what circumstances. Examples of policy requirements include permitting only necessary Internet Protocol (IP) protocols to pass, appropriate source and destination IP addresses to be used, particular Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) ports to be accessed, and certain Internet Control Message Protocol (ICMP) types and codes to be used. Generally, all inbound and outbound traffic not expressly permitted by the firewall policy should be blocked because such traffic is not needed by the organization. This practice reduces the risk of attack and can also decrease the volume of traffic carried on the organization's networks.

Identify all requirements that should be considered when determining which firewall to implement.

There are many considerations that organizations should include in their firewall selection and planning processes. Organizations need to determine which network areas need to be protected, and which types of firewall technologies will be most effective for the types of traffic that require protection. Several important performance considerations also exist, as well as concerns regarding the integration of the firewall into existing network and security infrastructures. Additionally, firewall solution design involves requirements relating to physical environment and personnel as well as consideration of possible future needs, such as plans to adopt new IPv6 technologies or virtual private networks (VPN).

Create rulesets that implement the organization's firewall policy while supporting firewall performance.

Firewall rulesets should be as specific as possible with regards to the network traffic they control. To create a ruleset involves determining what types of traffic are required, including protocols the firewall may need to use for management purposes. The details of creating rulesets vary widely by type of firewall and specific products, but many firewalls can have their performance improved by optimizing firewall rulesets. For example, some firewalls check traffic against rules in a sequential manner until a match is found; for these firewalls, rules that have the highest chance of matching traffic patterns should be placed at the top of the list wherever possible.

Manage firewall architectures, policies, software, and other components throughout the life of the firewall solutions.

There are many aspects to firewall management. For example, policy rules may need to be updated as the organization's requirements change, such as when new applications or hosts are implemented within the network. Firewall component performance also needs to be monitored to enable potential resource issues to be identified and addressed before components become overwhelmed. Logs and alerts should also be continuously monitored to identify threats—both successful and unsuccessful. Changes to firewall rulesets and policies should be managed by a formal process because of their potential to impact security, with ruleset reviews or tests performed periodically to ensure continued compliance with the organization's policies.

1. Introduction

1.1 Authority

The National Institute of Standards and Technology (NIST) developed this document in furtherance of its statutory responsibilities under the Federal Information Security Management Act (FISMA) of 2002, Public Law 107-347.

NIST is responsible for developing standards and guidelines, including minimum requirements, for providing adequate information security for all agency operations and assets; but such standards and guidelines shall not apply to national security systems. This guideline is consistent with the requirements of the Office of Management and Budget (OMB) Circular A-130, Section 8b(3), "Securing Agency Information Systems," as analyzed in A-130, Appendix IV: Analysis of Key Sections. Supplemental information is provided in A-130, Appendix III.

This guideline has been prepared for use by Federal agencies. It may be used by nongovernmental organizations on a voluntary basis and is not subject to copyright, though attribution is desired.

Nothing in this document should be taken to contradict standards and guidelines made mandatory and binding on Federal agencies by the Secretary of Commerce under statutory authority, nor should these guidelines be interpreted as altering or superseding the existing authorities of the Secretary of Commerce, Director of the OMB, or any other Federal official.

1.2 Purpose and Scope

This document seeks to assist organizations in understanding the capabilities of firewall technologies and firewall policies. It provides practical, real-world guidance on developing firewall policies and selecting, configuring, testing, deploying, and managing firewalls.

1.3 Audience

This document has been created primarily for technical information technology (IT) personnel such as network, security, and system engineers and administrators who are responsible for firewall design, selection, deployment, and management. Other IT personnel with network and system security responsibilities may also find this document to be useful. The content assumes some basic knowledge of networking and network security.

1.4 Document Structure

The remainder of this document is organized into four major sections:

- Section 2 provides an overview of a number of firewall technologies—including packet filtering, stateful inspection, and application-proxy gatewaying—and also provides information on host-based firewalls, personal firewalls, VPNs, and several types of specialized firewalls.
- Section 3 discusses the placement of firewalls within network architectures.
- Section 4 discusses firewall policies, and makes recommendations on the types of traffic that should be specified as prohibited.
- Section 5 provides an overview of firewall planning and implementation. It lists factors to consider when selecting firewall solutions, and provides recommendations for firewall configuration, testing,

deployment, and management.

The document also contains several appendices with supporting material:

- Appendices A and B contain a glossary and an acronym and abbreviation list, respectively.
- Appendix C lists print and online resources that may be of use in gaining a better understanding of firewalls.

2. Overview of Firewall Technologies

Firewalls are devices or programs that control the flow of network traffic between networks or hosts that employ differing security postures. While firewalls are often discussed in the context of Internet connectivity, they may also have applicability in other network environments. For example, many enterprise networks employ firewalls to restrict connectivity to and from the internal networks used to service more sensitive functions, such as accounting or personnel. By employing firewalls to control connectivity to these areas, an organization can prevent unauthorized access to its systems and resources. Inclusion of a proper firewall provides an additional layer of security.

Several types of firewall technologies are available. One way of comparing their capabilities is to look at the Transmission Control Protocol/Internet Protocol (TCP/IP) layers that each is able to examine. TCP/IP communications are composed of four layers that work together to transfer data between hosts. When a user wants to transfer data across networks, the data is passed from the highest layer through intermediate layers to the lowest layer, with each layer adding more information. The lowest layer sends the accumulated data through the physical network, with the data then passed upwards through the layers to its destination. Simply put, the data produced by a layer is encapsulated in a larger container by the layer below it. The four TCP/IP layers, from highest to lowest, are shown in Figure 2-1.

Application Layer. This layer sends and receives data for particular applications, such as Domain Name System (DNS), Hypertext Transfer Protocol (HTTP), and Simple Mail Transfer Protocol (SMTP).
Transport Layer. This layer provides connection-oriented or connectionless services for transporting application layer services between networks, and can optionally ensure communications reliability. TCP and UDP are commonly used transport layer protocols.
IP Layer (also known as the Network Layer). This layer routes packets across networks. Internet Protocol version 4 (IPv4) is the fundamental network layer protocol for TCP/IP. Other commonly used protocols at the network layer are Internet Protocol version 6 (IPv6), ICMP, and Internet Group Management Protocol (IGMP).
Hardware Layer (also known as the Data Link Layer). This layer handles communications on the physical network components. The best known data link layer protocol is Ethernet.

Figure 2-1. TCP/IP Layers

Addresses at the data link layer, which are assigned to network interfaces, are referred to as media access control (MAC) addresses—an example of this is an Ethernet address that belongs to an Ethernet card. Firewall policies rarely concern themselves with the data link layer. Addresses at the network layer are referred to as IP addresses. The transport layer identifies specific network applications and communication sessions as opposed to network addresses; a host may have any number of transport layer sessions with other hosts on the same network. The transport layer also includes the notion of *ports*—a destination port number generally identifies a service listening on the destination host, and a source port usually identifies the port number on the source host that the destination host should reply to. This combination of source IP address and port with destination IP address and port helps define the session. The highest layer represents end user applications—firewalls can inspect application traffic and use it as the basis for policy decisions.

Basic firewalls operate on one or a few layers—typically the lower layers—while more advanced firewalls examine all of the layers shown in Figure 2-1. Those that examine more layers can perform more granular and thorough examinations. Firewalls that understand the application layer can potentially accommodate advanced applications and protocols and provide services that are user-oriented. For example, a firewall that only handles lower layers cannot usually identify specific users, but a firewall

with application layer capabilities can enforce user authentication and log events to specific users.

2.1 Firewall Technologies

This section of the publication provides an overview of firewall technologies and basic information on the capabilities of several commonly used types. Firewalls are often combined with other systems—most notably routers—and many technologies often associated with firewalls are more accurately part of these other systems. For example, network address translation (NAT) is sometimes thought of as a firewall technology, but it is actually a routing technology.

2.1.1 Packet Filtering

The most basic feature of a firewall is the *packet filter*. Firewalls that are only packet filters—also known as *stateless inspection firewalls*—are essentially routing devices that provide access control functionality for host addresses and communication sessions. Unlike more advanced filters, packet filters are not concerned about the content of packets. Their access control functionality is governed by a set of directives referred to as a *ruleset*. Packet filtering capabilities are built into most operating systems and devices capable of routing; the most common example of a pure packet filtering device is a network router that employs access control lists.

In their most basic form, firewalls with packet filters operate at the network layer. This provides network access control based on several pieces of information contained in a packet, including:

- The packet's source IP address—the address of the host from which the network packet originated (such as 192.168.1.1)
- The packet's destination address—the address of the host the network packet is trying to reach (e.g., 192.168.2.1)
- The network protocol being used to communicate between source and destination hosts, such as TCP, UDP, or ICMP
- Possibly some characteristics of the transport layer communications sessions, such as session source and destination ports (e.g., TCP 80 for the destination port belonging to a Web server, TCP 1320 for the source port belonging to a personal computer accessing the server)
- The interface being traversed by the packet, and its direction (inbound or outbound).

Firewalls that are only packet filters and provide no advanced features have two main strengths—speed and flexibility. Since packet filters seldom examine data above the network layer (with the possible exception of limited transport layer information), they can operate very quickly. And because most modern network protocols can be accommodated via the network layer and below, packet filters can be used to provide some security for nearly any type of network communication or protocol. This simplicity allows firewalls with packet filters to be deployed into nearly any enterprise network infrastructure, and their speed and flexibility makes them ideal for placement at the outermost boundary, or perimeter, of an untrusted network to block incoming traffic—a procedure known as *ingress filtering*. Such packet filters, referred to as *boundary routers*, can block certain low-layer attacks, perform simple access control according to the policy in place (such as blocking unwanted protocols and permitting others), and pass permitted incoming traffic to more powerful firewalls that handle access control and filtering at higher layers. By performing this basic filtering, the boundary router reduces processing demands on the other firewalls. Figure 2-2 shows a packet filter in use as a boundary router.

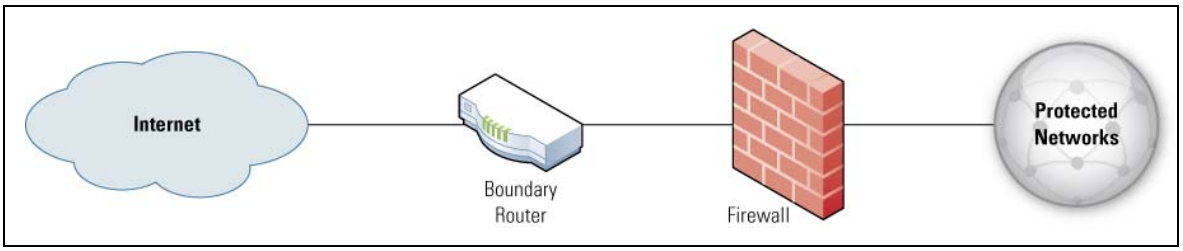

Figure 2-2. Packet Filter Used as Boundary Router

Outgoing traffic can also be filtered, a process referred to as *egress filtering*. Here, organizations can implement restrictions on their internal traffic, such as blocking the use of external file transfer protocol (FTP) servers or preventing denial of service (DoS) attacks from being launched from within the organization against outside entities. Organizations should only permit outbound traffic that uses the source IP addresses in use by the organization—a process that helps block traffic with spoofed addresses from leaking onto other networks. Spoofed addresses can be caused by malicious events such as malware infections or compromised hosts being used to launch attacks, or by inadvertent misconfigurations.

Firewalls that solely utilize packet filters offer speed and flexibility, but they also have built-in limitations. Because packet filters do not examine upper layer data, they are unable to prevent attacks that employ application-specific vulnerabilities or functions. For example, a packet filter firewall cannot block specific application commands—if a packet filter allows a specific application, all functions available within that application will also be permitted. The inability to examine upper layer data also prevents the support of advanced user authentication schemes and limits the value of logging, as most logs contain the same information used to make access control decisions (source address, destination address, and traffic type).

Packet filters are generally vulnerable to attacks and exploits that take advantage of problems within the TCP/IP specification and protocol stack. For example, many packet filters are unable to detect when a packet's network layer addressing information has been spoofed or otherwise altered. Spoofing attacks are generally employed by intruders to bypass the security controls implemented in a firewall platform. Firewalls that operate at higher layers can thwart some spoofing attacks by verifying that a session is established, or by authenticating users before allowing traffic to pass.

Because of these limitations, firewalls with only packet filters are typically used as the first line of defense at a network's perimeter to provide basic traffic filtering. Firewalls with additional capabilities—which may not be able to handle packets as quickly as a packet filter—are typically placed behind it. Alternatively, firewalls with advanced capabilities can be used at the perimeter if they are fast enough to handle incoming and outgoing traffic.

2.1.2 Stateful Inspection

Stateful inspection improves on the functions of packet filters by tracking the state of connections and blocking packets that deviate from the expected state. This is accomplished by incorporating greater awareness of the transport layer. As with packet filtering, stateful inspection intercepts packets at the network layer and inspects them to see if they are permitted by an existing firewall rule, but unlike packet filtering, stateful inspection keeps track of each connection in a state table. While the details of state table entries vary by firewall product, they typically include source IP address, destination IP address, port numbers, and connection state information.

Three major states exist for TCP traffic—connection establishment, usage, and termination (which refers to both an endpoint requesting that a connection be closed and a connection with a long period of inactivity.) Stateful inspection in a firewall examines certain values in the TCP headers to monitor the state of each connection. Each new packet is compared by the firewall to the firewall's state table to determine if the packet's state contradicts its expected state. For example, an attacker could generate a packet with a header indicating it is part of an established connection, in hopes it will pass through a firewall. If the firewall uses stateful inspection, it will first verify that the packet is part of an established connection listed in the state table.

Table 2-1 provides an example of a state table. If a device on the internal network (shown here as 192.168.1.100) attempts to connect to a device outside the firewall (192.0.2.71), the connection attempt is first checked to see if it is permitted by the firewall ruleset. If it is permitted, an entry is added to the state table that indicates a new session is being initiated, as shown in the first entry under "Connection State" in Table 2-1. If 192.0.2.71 and 192.168.1.100 complete the three-way TCP handshake, the connection state will change to "established" and all subsequent traffic matching the entry will be allowed to pass through the firewall.

Table 2-1. State Table Example

Source Address	Source Port	Destination Address	Destination Port	Connection State
192.168.1.100	1030	192.0.2.71	80	Initiated
192.168.1.102	1031	10.12.18.74	80	Established
192.168.1.101	1033	10.66.32.122	25	Established
192.168.1.106	1035	10.231.32.12	79	Established

Because some protocols, most notably UDP, are connectionless and do not have a formal process for initializing, establishing, and terminating a connection, state cannot be established at the transport layer as it is for TCP. For these protocols, most firewalls with stateful inspection are only able to track the source and destination IP addresses and ports. UDP packets must still match an entry in the state table based on source and destination IP address and port information to be permitted to pass—but in such a system, a DNS response from an external source would be permitted to pass only if the firewall had previously seen a corresponding DNS query from an internal source. Since the firewall is unable to determine when a session has ended, the entry is removed from the state table after a preconfigured timeout value is reached. Application-level firewalls that are able to recognize DNS over UDP will terminate a session after a DNS response is received.

A newer trend in stateful inspection is the addition of a stateful protocol analysis capability, referred to by some vendors as *deep packet inspection.*[1] Stateful protocol analysis improves upon standard stateful inspection through adding basic intrusion detection technology—an inspection engine that analyzes protocols at the network, transport, and application layers to compare vendor-developed profiles of benign protocol activity against observed events to identify deviations. This enables the identification of unexpected sequences of commands, such as issuing the same command repeatedly or issuing a command that was not preceded by another command on which it is dependent. These suspicious commands often originate from buffer overflow attacks, DoS attacks, malware, and other forms of attack carried out within

[1] This publication uses the term *stateful protocol analysis* because it is appropriate for analyzing both network-based and host-based activity, whereas *deep packet inspection* is an appropriate term for network-based activity only. Historically, there has not been consensus in the security community as to the meaning of deep packet inspection.

application protocols such as HTTP. Another common feature is reasonableness checks for individual commands, such as minimum and maximum lengths for arguments. For example, a username argument with a length of 1000 characters is suspicious—even more so if it contains binary data.

Firewalls with both stateful inspection and stateful protocol analysis capabilities are not full-fledged intrusion detection and prevention systems (IDPS), which usually offer much more extensive attack detection and prevention capabilities. For example, IDPSs also use signature-based and/or anomaly-based analysis to detect additional problems within network traffic.[2]

2.1.3 Application-Proxy Gateways

An *application-proxy gateway* is a feature of advanced firewalls that combines lower layer access control with upper layer functionality. These firewalls contain a proxy agent that acts as an intermediary between two hosts that wish to communicate with each other, and never allows a direct connection between the two hosts. Each successful connection attempt actually results in the creation of two separate connections—one between the client and the proxy server, and another between the proxy server and the true destination (shown in Figure 2-3). The proxy is meant to be transparent to the two hosts, and from their perspectives there appears to be a direct connection. Because external hosts only communicate with the proxy agent, internal IP addresses are not made known to the outside world. The proxy agent interfaces directly with the firewall ruleset to determine whether a given piece of network traffic should be allowed to transit the firewall. In addition to the ruleset, each proxy agent has the ability to require authentication of each individual network user. This user authentication can take many forms, including user ID and password, hardware or software token, source address, and biometrics.

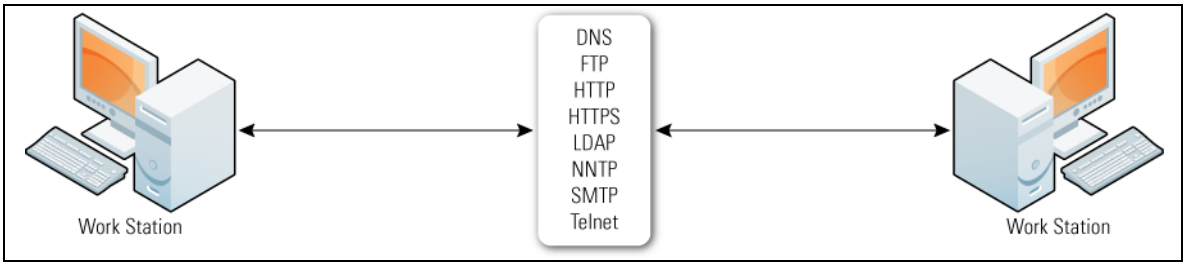

Figure 2-3. Typical Proxy Agents

The proxy gateway operates at the application layer and can inspect the actual content of the traffic. Unlike stateful protocol analysis, which mainly verifies that traffic is consistent with protocol definitions, application-proxy gateways break down the data and more thoroughly examine packet content—distinguishing between normal traffic for a specific protocol and traffic that could contain exploits for known flaws. These gateways also perform the TCP handshake with the source system and are able to protect against exploitations at each step of a communication. In addition, gateways can make decisions to permit or deny traffic based on information in the application protocol headers or payloads. For instance, a gateway can determine if an email message contains a certain type of attachment that the organization does not permit (such as an executable file), or if instant messaging (IM) is being used over port 80 (typically used for HTTP). Another feature of the gateway is that it can restrict specific actions from being performed (e.g., users could be prevented from using the FTP "put" command, which allows users to write files to the FTP server). It can also be used to allow or deny Web pages that contain

[2] For additional information about IDPS, see NIST Special Publication (SP) 800-94, *Guide to Intrusion Detection and Prevention Systems (IDPS)* (http://csrc.nist.gov/publications/nistpubs/).

particular types of active content, such as Java or ActiveX. Once the gateway determines that data should be permitted, it is forwarded to the destination host.

Application-proxy gateways have numerous advantages over packet filters and stateful inspection. First, an application-proxy gateway offers a higher level of security because it prevents direct connections between two hosts and it inspects traffic content to identify policy violations. Second, these gateways usually have more extensive logging capabilities because they can examine an entire packet rather than just network addresses and ports—for example, application-proxy gateway logs can record application-specific commands from within the network traffic. Also, the user authentication capabilities inherent in application-proxy gateway architectures are superior. Another potential advantage is that some application-proxy gateways have the ability to decrypt packets (such as Secure Sockets Layer [SSL]-protected payloads), examine them, and re-encrypt them before sending them on to the proper destination host. Data that the gateway cannot decrypt is passed directly through to the application. Finally, application-proxy gateways are better able to detect address spoofing attacks.

The advanced functionality of firewalls with application-proxy gateways also has several disadvantages when compared to packet filtering and stateful inspection. First, because of the "full packet awareness" of application-proxy gateways, the firewall spends much more time reading and interpreting each packet. Because of this, some of these gateways are poorly suited to high-bandwidth or real-time applications—but application-proxy gateways rated for high bandwidth are available. To reduce the load on the firewall, a dedicated proxy server (discussed in Section 2.1.5) can be used to secure less time-sensitive services such as email and most Web traffic. Another disadvantage is that application-proxy gateways tend to be limited in terms of support for new network applications and protocols—an individual, application-specific proxy agent is required for each type of network traffic that needs to transit a firewall. Many application-proxy gateway firewall vendors provide generic proxy agents to support undefined network protocols or applications. Those generic agents tend to negate many of the strengths of the application-proxy gateway architecture because they simply allow traffic to "tunnel" through the firewall.

2.1.4 Circuit-Level Gateways

A *circuit-level gateway* is another type of proxy, and is sometimes referred to as a *circuit-level proxy*. Besides its proxy capabilities, which shield internal systems from the outside world, circuit-level gateways validate each connection before it is established in a manner similar to that of stateful inspection. When a connection request is received, the circuit-level gateway checks its ruleset to determine if the connection should be allowed. Rules are typically based upon the following—destination IP address and/or port, source IP address and/or port, and requested application protocol. Some circuit-level gateways can also base rules on user authentication or time restrictions.

Once a connection is permitted, an entry is placed in a virtual circuit table that also contains state information. Packets listed in the table are allowed to pass through the firewall without further validation. When the connection has been terminated or has been inactive for a pre-determined period of time, the entry is removed from the table. A circuit-level proxy provides many of the same features as a firewall that has stateful inspection, with the added functionality of a proxy to prevent direct connections between hosts on opposite sides of the firewall. Circuit-level gateways are usually faster than application-proxy gateway firewalls because they perform fewer evaluations on the data—they do not examine the content of the applications.

2.1.5 Dedicated Proxy Servers

Dedicated proxy servers differ from application-proxy and circuit-level gateways in that while they retain proxy control of traffic, they do not have firewalling capabilities. Although dedicated proxy servers are

not firewalls, they are described in this section because of their close relationship to application-proxy gateway firewalls and circuit-level gateway firewalls. Many proxies are application-specific, and some actually perform analysis and validation of common application protocols such as HTTP. Because these servers do not have firewalling capabilities, they are typically deployed behind traditional firewall platforms. Typically, a main firewall could accept inbound traffic, determine which application is being targeted, and hand off traffic to the appropriate proxy server (e.g., email proxy). This server would perform filtering or logging operations on the traffic, then forward it to internal systems. A proxy server could also accept outbound traffic directly from internal systems, filter or log the traffic, and pass it to the firewall for outbound delivery. An example of this is an HTTP proxy deployed behind the firewall—users would need to connect to this proxy en route to connecting to external Web servers. Dedicated proxy servers are generally used to decrease firewall workload and conduct specialized filtering and logging that might be difficult to perform on the firewall itself.

In recent years, the use of *inbound* proxy servers has decreased dramatically. This is because an inbound proxy server must mimic capabilities of the real server it is protecting, which becomes nearly impossible when protecting a server with many features. Using a proxy server with fewer capabilities than the server it is protecting renders the non-matched capabilities unusable. Additionally, the essential features that inbound proxy servers should have (logging, access control, and so on) are usually built into the real servers. Most proxy servers now in use are *outbound* proxy servers, with the most common being HTTP proxies.

Figure 2-4 shows a sample diagram of a network employing dedicated proxy servers for HTTP and email that have been placed behind another firewall system. Here, the email proxy could be the organization's mail gateway for inbound and outbound email—not really a proxy at all, but a full-fledged mail server. All messages and communications must go through the proxy before they can be forwarded to other internal mail servers. Breaking the direct line of communication between the Internet and the internal mail servers makes it much more difficult to attack those mail servers; only the mail gateway can be attacked directly. The HTTP proxy would handle outbound connections to external Web servers and possibly filter for active content. Many organizations enable caching of frequently used Web pages on the proxy to reduce network traffic and improve response times.

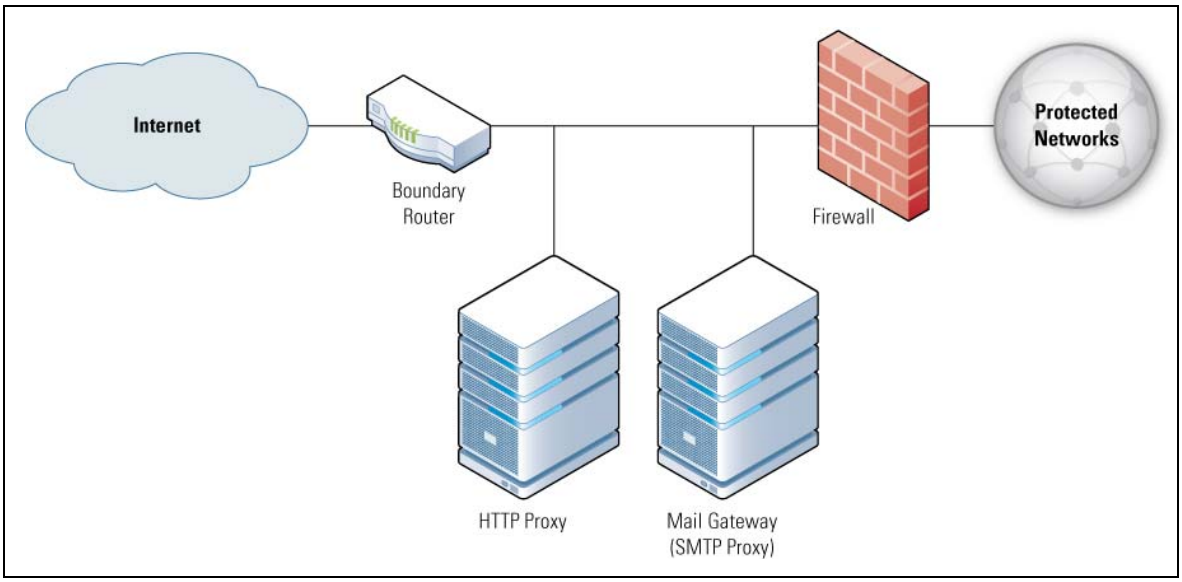

Figure 2-4. Application Proxy Configuration

2.1.6 Virtual Private Networking

Firewall devices at the edge of a network are sometimes required to do more than block unwanted traffic. A common requirement for these firewalls is to encrypt and decrypt specific network traffic flows between the protected network and external networks. This nearly always involves virtual private networks (VPN), which use additional protocols to encrypt traffic and provide user authentication and integrity checking. VPNs are most often used to provide secure network links across untrusted networks. For example, VPN technology is widely used to extend the protected network of a multi-site organization across the Internet, and sometimes to provide secure remote user access to internal organizational networks via the Internet. Two common choices for secure VPNs are Internet Protocol Security (IPsec)[3] and Secure Sockets Layer (SSL)/Transport Layer Security (TLS).[4]

The three most common VPN architectures are gateway-to-gateway, host-to-gateway, and host-to-host.[5] Gateway-to-gateway architectures are the most widely used, and connect multiple fixed sites over public lines through the use of VPN gateways—for example, to connect branch offices to an organization's headquarters. A VPN gateway is usually part of another network device such as a firewall or router. When a VPN connection is established between the two gateways, users at branch locations are unaware of the connection and do not require any special settings on their computers. The second type of architecture, host-to-gateway, provides a secure connection to the network for individual users, usually called *remote users*, who are located outside of the organization (at home, in a hotel, etc.) Here, a client on the user machine negotiates the secure connection with the organization's VPN gateway. The third architecture, host-to-host, is the least commonly used. This setup typically enables remote administration of a single server.

For gateway-to-gateway and host-to-gateway VPNs, the VPN functionality is often part of the firewall itself. Placing it behind the firewall would require VPN traffic to be passed through the firewall while encrypted, preventing the firewall from inspecting the traffic.

All VPNs allow the firewall administrator to decide which users have access to which network resources. This access control is normally on a per-user basis; that is, the VPN policy outlines which users are authorized to access which resources. VPNs generally rely on authentication protocols such as RADIUS (Request for Comment [RFC] 2865, *Remote Authentication Dial In User Service*). RADIUS uses several different types of authentication credentials, with the most common examples being username and password, digital signatures, and hardware tokens.

To run VPN functionality on a firewall requires additional resources that depend upon the amount of traffic flowing across the VPN and the type of encryption being used. For some environments, the added traffic associated with VPNs might require additional capacity planning and resources. Planning is also needed to determine the type of VPN (gateway-to-gateway and/or remote access) that should be included in the firewall.

2.2 Locations for Firewalls

Although firewalls at a network's perimeter provide some measure of protection for internal hosts, in many cases additional network protection is required. Network firewalls are not able to recognize all instances and forms of attack, allowing some attacks to penetrate and reach internal hosts—and attacks

---

[3] For additional information on IPsec, see NIST SP 800-77, *Guide to IPsec VPNs* (http://csrc.nist.gov/publications/nistpubs/).

[4] For additional information on SSL and TLS, see NIST SP 800-52, *Guidelines for the Selection and Use of Transport Layer Security (TLS) Implementations* and NIST SP 800-113, *Guide to SSL VPNs* (http://csrc.nist.gov/publications/nistpubs/).

[5] For additional information on VPN architectures, see NIST SP 800-77, *Guide to IPsec VPNs* and NIST SP 800-113, *Guide to SSL VPNs* (http://csrc.nist.gov/publications/nistpubs/).

sent from one internal host to another may not even pass through a network firewall. Because of these and other factors, network designers often include firewall functionality at places other than the network perimeter to provide an additional layer of security.

2.2.1 Host-Based Firewalls and Personal Firewalls

To better protect hosts from network-based attacks, host-based firewalls for servers and personal firewalls for desktop and laptop computers provide an additional layer of security. These firewalls are software-based, residing on the hosts they are protecting—and each monitors and controls the incoming and outgoing network traffic for a single host. They can also provide more granular protection than network firewalls to meet the needs of specific hosts.

Host-based firewalls are available as part of server operating systems such as Linux and Windows, and can sometimes also be installed as third party add-ons under various operating systems. Configuring a host-based firewall to allow only necessary traffic to the server provides protection against malicious activity from all hosts, including those on the same subnet or on other internal subnets not separated by a network firewall. Limiting outgoing traffic from a server may also be helpful in preventing certain malware that infects a host from spreading to other hosts.[6] Host-based firewalls usually perform logging, and can often be configured to perform address-based access controls.

A personal firewall is software that runs on a desktop or laptop computer with a user-focused operating system such as Windows or Macintosh OS X. A personal firewall is similar to a host-based firewall, but because the computer being protected is meant for end users, the interface is usually different (and presumably easier for the typical user to understand). They provide an additional layer of security for desktop and laptop computers located both inside and outside perimeter firewalls (e.g., mobile laptop users); can restrict inbound communications; and can often limit outbound communications. This not only allows personal firewalls to protect desktops and laptops from incoming attacks, but also limits the spread of malware from infected desktops and laptops and the downloading and use of unauthorized software such as peer-to-peer file sharing utilities. Personal firewalls are often packaged with antivirus programs, intrusion detection software, and other security utilities.[7]

Personal firewalls should be configured to permit only the types of communications that are permitted by the organization's security policy and to deny all other communications. Many personal firewalls can be configured to allow communications based on lists of authorized applications—such as Web browsers contacting Web servers and email clients sending and receiving email messages—and to deny communications involving any other applications.

Management of personal firewalls should be centralized if at all possible to help efficiently create, distribute, and enforce policies for all users and groups. Doing this will ensure that the organization's security policy will be in effect whenever a user is accessing the organization's computing resources. But regardless of whether a personal firewall is managed by central administrators or individual users, any warning messages should be shared with users to help them rectify problems that are found.

---

[6] If an attacker compromises a host and gains administrator-level privileges, the attacker can disable or circumvent the host-based firewall.

[7] For additional information about personal firewalls, see NIST SP 800-114, *User's Guide to Securing External Devices for Telework and Remote Access*, (http://csrc.nist.gov/publications/nistpubs/).

2.2.2 Personal Firewall Appliances

In addition to using personal firewalls on their personal computers (PC), some teleworkers also use a small, inexpensive device called a firewall appliance or firewall router to protect the computers on their home networks. A personal firewall appliance performs functions similar to a personal firewall, including some of the more advanced features listed earlier in this section—such as VPN. Even if each computer on a home network is using a personal firewall, a firewall appliance is still a valuable added layer of security. Should a personal firewall on a computer malfunction, be disabled, or be misconfigured, the firewall appliance can still protect the computer from unauthorized network communications from external computers.[8]

2.2.3 Distributed Firewalling

Distributed firewalling is an emerging security technology in firewall deployment that moves security from the perimeter to device endpoints. This is accomplished by placing a firewall in or directly in front of every endpoint and other appropriate devices in the network. The theory of distributed firewalling is that this can ease the burden on the perimeter and internal firewalls, which have traditionally been major chokepoints for network access. As distributed firewall technology evolves, it could mostly or completely remove the need for perimeter and internal firewalls. However, few organizations are presently deploying distributed firewalling.

Distributed firewalls rely on a central policy server that pushes security policies out to the firewall residing on each device. Distributed firewalls expand on the concept of centrally managing personal and host-based firewalls by giving each device a certificate that identifies it to other devices on the network. These certificates can also be used to identify which machines have rights to certain resources—for example, an internal Web server could have a security policy that only grants incoming access to workstations with a specific certificate.

This concept provides access control between workstations (both inside and outside the organization's intranet) and internal resources, allowing the network to scale well in an environment that is largely made up of mobile users. To help prevent spoofing, the distributed firewall system should implement authentication measures that ensure the proper identity of each endpoint on the network. Otherwise, an attacker may be able to gain access to certificates and acquire unauthorized access to network resources.

2.3 Function-Specific Firewalls

Firewalls are sometimes implemented to protect special-purpose systems, applying firewall concepts to technologies such as private branch exchange (PBX) systems. Others take existing firewall technology, such as stateful protocol analysis and application filtering, and apply them to a particular protocol or technology—such as Extensible Markup Language (XML) or email. These firewalls are frequently paired with antivirus technology and intrusion detection system (IDS) capabilities that are specific to the protocol or technology. Examples of these types of firewalls are provided below.

2.3.1 PBX Firewall

Traditionally, PBX resources have been managed using text terminals or proprietary management consoles. But in recent times it has become common for PBX vendors to include management software that requires network layer connectivity, especially for the newer generation of smaller, modular PBX

[8] Additional information on personal firewall appliances is available from NIST SP 800-114, *User's Guide to Securing External Devices for Telework and Remote Access* (http://csrc.nist.gov/publications/nistpubs/).

systems. It is not uncommon for newer PBX systems to implement modularity through the use of network layer network connections between PBX nodes.

A PBX firewall typically provides functionality similar to that of an Internet firewall—i.e., enforcing a user-specified security policy on telephone line use within an organization. For example, the firewall may enforce the following rules on a set of lines:

1. Always allow emergency (911) calls
2. Disallow incoming modem calls
3. Disallow outgoing modem calls
4. Allow all other traffic.

Like packet filtering firewalls, a PBX firewall works by filtering calls according to characteristics such as call direction (inbound or outbound), call source telephone number, call destination telephone number, call type (e.g., emergency, 1-800), and start time. Administrators may be provided with options to log these or other characteristics of calls, block certain types of calls, or issue a real-time alert when a designated call rule is violated.

Using a firewall to regulate access to PBX resources creates an additional audit trail—both the PBX and the firewall log the management session.

2.3.2 XML Firewall

XML firewalls, also known as *XML gateways*, are designed to examine all content that is part of XML messaging. These firewalls usually reside in front of servers offering Web services, and typically examine SOAP headers and XML message bodies to determine if the traffic should be allowed or blocked. They can also provide authentication, access control, encryption, signature verification, logging, and alerting, and look for malformed messages and other malicious content.[9]

2.3.3 Email Firewall

Many vendors offer firewalls specifically to protect email services. These devices are often designed to inspect incoming email for embedded malware, as well as scan the bodies of email messages for indicators of spam and phishing attacks. They can also block incoming traffic from known spam sites, and track email traffic for abnormal patterns such as large traffic volume. In addition, email firewalls may offer additional control for outbound messages—for example, scanning email to see if it contains information that may be in violation of the organization's policies. These firewalls may also offer additional security features such as encryption and extensive logging.[10]

---

[9] For additional information about XML and XML firewalls, see NIST SP 800-95, *Guide to Secure Web Services* (http://csrc.nist.gov/publications/nistpubs/).

[10] For additional information about email security, see NIST SP 800-45 Version 2, *Guidelines on Electronic Mail Security* (http://csrc.nist.gov/publications/nistpubs/).

3. Firewalls and Network Architectures

Firewalls are used to separate networks with differing security requirements, such as the Internet and an internal network that houses servers with sensitive data. Organizations should use firewalls wherever their internal networks and systems interface with external networks and systems, and where security requirements vary among their internal networks. This section is intended to help organizations determine where firewalls should be placed, and where other networks and systems should be located in relation to the firewalls.

Since one of the primary functions of a firewall is to prevent unwanted traffic from entering a network (and, in some cases, from exiting it), firewalls should be placed at the edge of logical network boundaries. This normally means that firewalls are positioned either as a node where the network splits into multiple paths, or inline along a single path. In routed networks, the firewall usually resides just before the ingress to the router—and is sometimes co-resident with the router. It is rare to place the firewall for a multi-path node after the router because the firewall device would need to watch each of the multiple exit paths that typically exist in such situations. The vast majority of hardware firewall devices contain router capabilities, and in switched networks, a firewall is often part of the switch itself to enable it to protect as many of the switched segments as possible.

Firewall vendors often vary in their terminology for the logical flow of firewall traffic. A firewall takes traffic that has not been checked, checks it against the firewall's policy, and then acts accordingly (e.g., passes the traffic, blocks it, passes it with some modification). Because all traffic on a network has a direction, policies are based on the direction that the traffic is moving. For the purposes of this document, traffic that has not yet been checked is coming from the “unprotected side” of the firewall and is moving towards the “protected side.” Some firewalls check traffic in both directions—for example, if they are set up to prevent specific traffic from an organization's LAN from escaping to the Internet. In these cases, the protected side of the firewall is the one facing the outside network.

Section 2 lists many different types of firewalls. Network firewalls are almost always hardware devices with multiple network interfaces; host-based and personal firewalls involve software that resides on a single computer and protects only that computer; and personal firewall appliances are designed to protect a single PC or a small office/home office network. This section focuses on network firewalls because the other types have no network topology issues.

3.1 Network Layouts with Firewalls

Figure 3-1 shows a typical network layout with a hardware firewall device acting as a router. The unprotected side of the firewall connects to the single path labeled “WAN,” and the protected side connects to three paths labeled “LAN1,” “LAN2,” and “LAN3.” The firewall acts as a router for traffic between the Wide Area Network (WAN) path and the Local Area Network (LAN) paths.

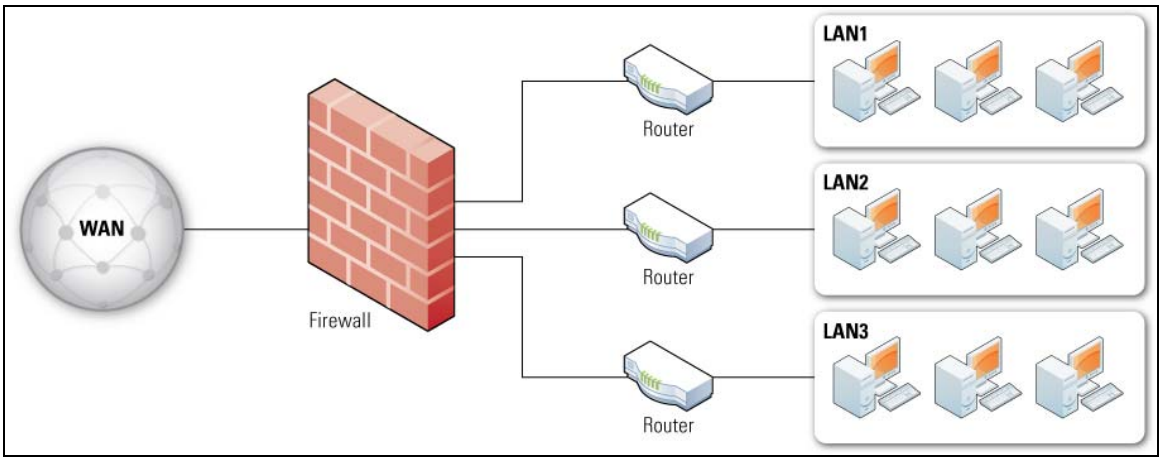

Figure 3-1. Simple Routed Network with Firewall Device

Many hardware firewall devices have a feature called *DMZ*, an acronym related to the demilitarized zones that are sometimes set up between warring countries. While no single technical definition exists for firewall DMZs, they are usually interfaces on a routing firewall that are similar to the interfaces found on the firewall's protected side. The major difference is that traffic moving between the DMZ and other interfaces on the protected side of the firewall still goes through the firewall and can have firewall protection policies applied.

An example of this is Figure 3-2, a simple network layout of a firewall with a DMZ. Traffic from the Internet goes into the firewall, and is routed to systems on the firewall's protected side or to systems on the DMZ. Traffic between systems on the DMZ and systems on the protected network goes through the firewall, and can optionally have firewall policies applied.

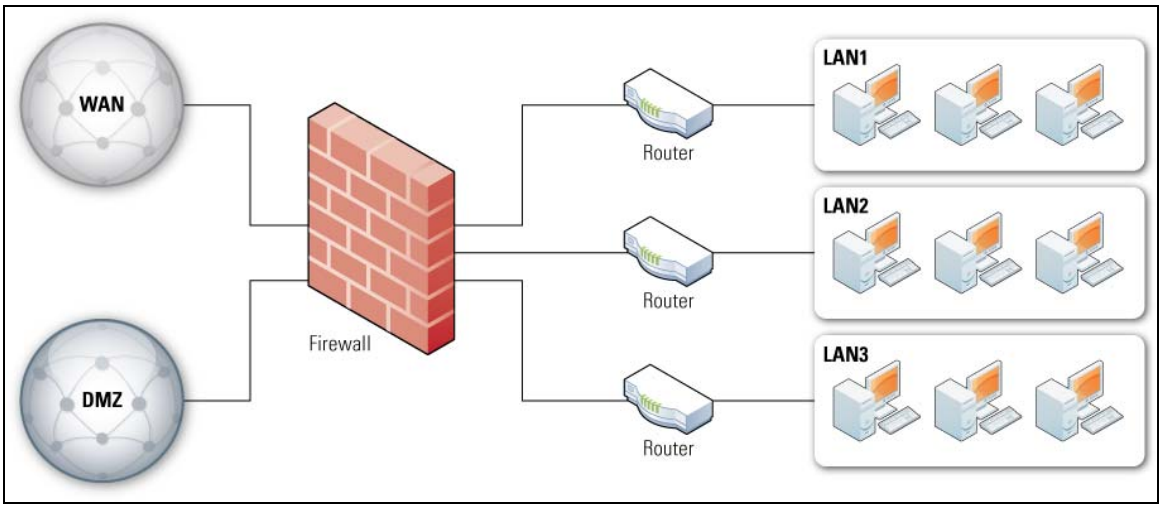

Figure 3-2. Firewall with a DMZ

Most network architectures are hierarchical, meaning that a single path from an outside network splits into multiple paths on the inside network—and it is generally most efficient to place a firewall at the node where the paths split. This has the advantage of positioning the firewall where there is no question as to what is "outside" and what is "inside." However, there may be reasons to have additional firewalls on the inside of the network, such as to protect one set of computers from another.

If a network's architecture is not hierarchical, the same firewall policies should be used on all ingresses to the network. In many organizations, there is only supposed be one ingress to the network, but other ingresses are set up on an ad-hoc basis, often in ways that are not allowed by overall policy. In these situations, if a properly configured firewall is not placed at each entry point, malicious traffic that would normally be blocked by the main ingress can enter the network by other means.

The network in Figure 3-3 is similar to that shown in Figure 3-1, except that a user has added an unauthorized connection in LAN2. This connection might be an accidental wireless connection from a network run by a neighbor—or it may have been set up intentionally to avoid specific policies on the firewall. Regardless of the reason, the connection allows traffic that did not pass through the firewall to traverse the network and have access to LAN1, LAN2, and LAN3.

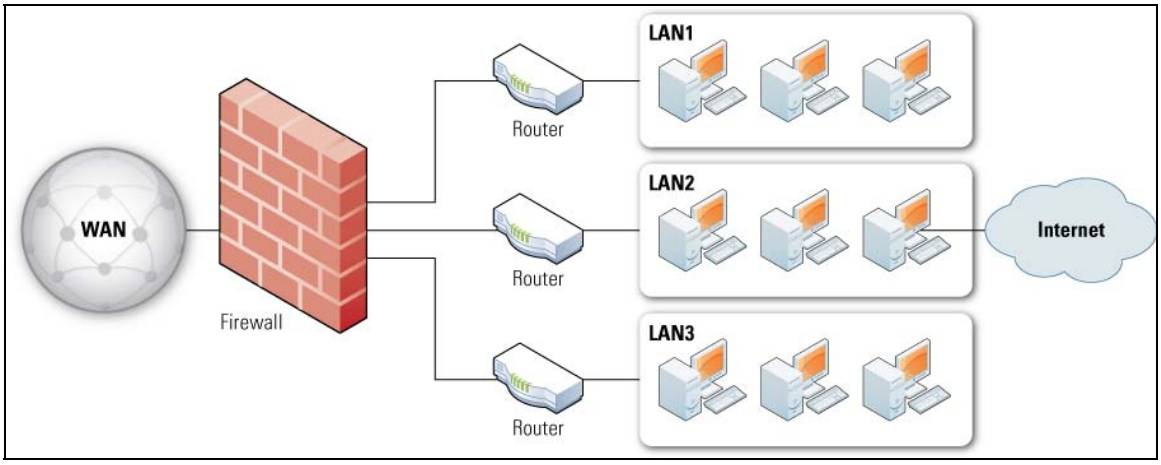

Figure 3-3. Network with a Second Connection from the Outside

3.2 Architecture with Multiple Layers of Firewalls

There is no limitation on where a firewall can be placed in a network. While firewalls should be at the edge of a logical network boundary, creating an "inside" and "outside" on either side of the firewall, a network administrator may wish to have additional boundaries within the network and deploy additional firewalls to establish such boundaries. The use of multiple layers of firewalls is quite common to provide defense-in-depth. An example of this was mentioned in Section 2.2.1, where a host-based firewall creates a boundary just before the host it is installed upon and adds another set of firewall policies to the architecture of the network. Using multiple layers of network firewalls is another common technique.

A typical situation that requires multiple layers of network firewalls is the presence of internal users with varying levels of trust. For example, an organization might want to protect its accounting databases from being accessed by users who are not part of the accounting department. This could be accomplished by placing one firewall at the edge of the network (to prevent general access to the network from the Internet) and another at the edge of the internal network that defines the boundary of the accounting department. The inner firewall would block access to the database server by anyone outside the accounting network while allowing limited access to other resources on the accounting network.

Another typical use for firewalls inside a network with a firewall at its edge involves visitors who need access to the Internet. Many organizations deploy specific wireless access points within their networks for visitor use. A firewall between each access point and the rest of the internal network can prevent visitors from accessing the local network with the same privileges as an employee.

Placing a firewall within a network that already has one at the edge requires good planning and policy coordination to prevent inadvertent security lapses. When designing policies for an inner firewall, the administrator could make assumptions that result in poor policy choices—for example, if the inner firewall's administrator assumes that the outer firewall is already preventing certain types of traffic from reaching the inner firewall, and the outer firewall's administrator later modifies existing policy, hosts behind the inner firewall will be exposed to additional threats. A good approach is to duplicate outer firewall policies that are also relevant for inner firewalls on each inner firewall. This may be difficult if these firewalls are not able to coordinate their policies automatically, which is particularly likely when firewalls are from different manufacturers.

Another common problem with using multiple layers of network firewalls is the increased difficulty it presents in tracing firewall problems. If one firewall stands between a user and a server, and the user cannot connect to the server, it is easy to check that firewall's logs to see if the connection is being permitted. But if multiple firewalls are involved, the problem becomes more difficult because an administrator must locate all firewalls in the chain and check their logs to find where the problem originates. The presence of multiple layers of application layer gateways (ALG) is particularly daunting, because each ALG can change a message, which makes debugging even more difficult.

4. Firewall Policy

A firewall policy dictates how firewalls should handle network traffic for specific IP addresses and address ranges, protocols, applications, and content types (e.g., active content) based on the organization's information security policies. Before a firewall policy is created, some form of risk analysis should be performed to develop a list of the types of traffic needed by the organization and categorize how they must be secured—including which types of traffic can traverse a firewall under what circumstances.[11] This risk analysis should be based on an evaluation of threats; vulnerabilities; countermeasures in place to mitigate vulnerabilities; and the impact if systems or data are compromised.

Firewall policy should be maintained and updated frequently as classes of new attacks or vulnerabilities arise, or as the organization's needs regarding network applications change. This should make the process of creating a firewall ruleset less error-prone and more verifiable, since the ruleset can be compared to the applications matrix. The policy should also include specific guidance on how to address changes to the ruleset.

Generally, firewalls should block all inbound and outbound traffic that has not been expressly permitted by the firewall policy—traffic that is not needed by the organization. This practice, known as "deny by default," decreases the risk of attack and can also reduce the volume of traffic carried on the organization's networks. Because of the dynamic nature of hosts, networks, protocols, and applications, deny by default is a more secure approach than permitting all traffic that is not explicitly forbidden.

The remainder of this section provides details on what types of traffic should be blocked. Section 4.1 discusses policies for packet filtering and stateful inspection based on IP addresses and other IP characteristics, while Section 4.2 covers policies relating to application-specific traffic.

4.1 Policies Based on IP Addresses and Protocols

Firewall policies should only allow necessary IP protocols through. Examples of commonly used IP protocols, with their IP protocol numbers,[12] are ICMP (1), TCP (6), and UDP (17). Other IP protocols, such as IPsec components Encapsulating Security Payload (ESP) (50) and Authentication Header (AH) (51) and routing protocols, may also need to pass through firewalls. These necessary protocols should be restricted whenever possible to the specific hosts and networks within the organization with a need to use them. By permitting only necessary protocols, all unnecessary IP protocols are denied by default.

4.1.1 IP Addresses and Other IP Characteristics

Firewall policies should only permit appropriate source and destination IP addresses to be used. Specific recommendations for IP addresses include:

- Traffic with invalid source or destination addresses should always be blocked, regardless of the firewall location. Examples of relatively common invalid IPv4 addresses are 127.0.0.1 (also known as the localhost address) and 0.0.0.0 (interpreted by some operating systems as a localhost or a broadcast address). These have no legitimate use on a network.

[11] The process to perform a risk assessment and create this type of list is not detailed here. For additional information, see NIST SP 800-30, *Risk Management Guide for Information Technology Systems*, and SP 800-18 Revision 1, *Guide for Developing Security Plans for Federal Information Systems*, at http://csrc.nist.gov/publications/nistpubs/.

[12] IP protocol number assignments are defined in http://www.iana.org/assignments/protocol-numbers.

- Traffic with an invalid source address for incoming traffic or destination address for outgoing traffic (an "external" address) should be blocked at the network perimeter. This traffic is often caused by malware, spoofing, denial of service attacks, or misconfigured equipment. Two types of invalid external addresses are:
  - An IPv4 address within the ranges in RFC 1918, *Address Allocation for Private Internets*, that are reserved for private networks. These ranges are 10.0.0.0 to 10.255.255.255 (10.0.0.0/8 in CIDR [Classless Inter-Domain Routing] notation), 172.16.0.0 to 172.31.255.255 (172.16.0.0/12), and 192.168.0.0 to 192.168.255.255 (192.168.0.0 /16).
  - An address that is not in an Internet Assigned Numbers Authority (IANA)-designated assigned range.[13] To help organizations in filtering invalid external IPv4 addresses, various organizations publish *bogon lists*—lists of address ranges that are not valid for Internet use.[14] Bogon lists from reputable organizations are based on close monitoring of IANA-assigned ranges. When using these lists for blocking traffic, it is extremely important to update them at least weekly—failing to do this will prevent users from being able to communicate with new, legitimate sites whose IP addresses were assigned from numbers that appeared on the older lists.
- Traffic with a private destination address for incoming traffic or source address for outgoing traffic (an "internal" address) should be blocked at the network perimeter. Perimeter devices can perform address translation services to permit internal hosts with private addresses to communicate through the perimeter, but private addresses should not be passed through the network perimeter.
- Incoming traffic with a destination address of the firewall itself should be blocked unless the firewall is offering services for incoming traffic that require direct connections—for example, if the firewall is acting as an application proxy.

Organizations should also block the following types of traffic at the perimeter:

- Traffic containing IP source routing information, which allows a system to specify the routes that packets will employ while traveling from source to destination. This could potentially permit an attacker to construct a packet that bypasses network security controls. IP source routing is rarely used on modern networks, and valid applications are even less common on the Internet.
- Traffic containing directed broadcast addresses, which are broadcast addresses that are not in the same subnet as the originator. Any system that responds to the directed broadcast will then send its response to the system specified by the source, rather than to the source system itself. These packets can be used to create huge "storms" of network traffic for denial of service attacks.

Firewalls at the network perimeter should block all incoming traffic to networks and hosts that should not be accessible from external networks. These firewalls should also block all outgoing traffic from the organization's networks and hosts that should not be permitted to access external networks. Deciding which addresses should be blocked is often one of the most time-consuming aspects of developing firewall IP policies. It is also one of the most error-prone, because the IP address associated with an undesired entity often changes over time.

---

[13] The list of assigned IPv4 address ranges is available at http://www.iana.org/assignments/ipv4-address-space, and the IPv6 assignments are listed at http://www.iana.org/assignments/ipv6-address-space.

[14] One source of bogon lists is at http://www.cymru.com/Bogons/index.html, which also contains additional information on IP address filtering.

4.1.2 IPv6

IPv6 is a new version of IP that is increasingly being deployed. Although IPv6's internal format and address length differ from those of IPv4, many other features remain the same—and some of these are relevant to firewalls. For the features that are the same between IPv4 and IPv6, firewalls should work the same. For example, blocking all inbound and outbound traffic that has not been expressly permitted by the firewall policy should be done regardless of whether or not the traffic has an IPv4 or IPv6 address.

As of this writing, some firewalls cannot handle IPv6 traffic at all; others are able to handle it but have limited abilities to filter IPv6 traffic; and still others can filter IPv6 traffic at the same level or quality used for IPv4 traffic. Every organization that has any IPv6 traffic coming into its internal network needs a firewall that is capable of filtering this kind of traffic. These firewalls should have the following capabilities:

- The firewall should be able to use IPv6 addresses in all filtering rules that use IPv4 addresses.
- The administrative interface should allow administrators to clone IPv4 rules to IPv6 addresses to make administration easier.
- If the firewall can filter based on DNS lookup of domain names, it needs to use AAAA (IPv6 address records) records in the same way as A records (those used for IPv4 addresses).
- The firewall needs to be able to filter ICMPv6, as specified in RFC 4890, *Recommendations for Filtering ICMPv6 Messages in Firewalls*.
- Many sites tunnel IPv6 packets in IPv4 packets. This is particularly common for sites experimenting with IPv6, because it is currently much easier to obtain IPv6 transit from a *tunnel broker* through a v6-to-v4 tunnel than to get native IPv6 transit from an Internet service provider (ISP). A number of ways exist to do this, and standards for tunneling are still evolving. If the firewall is able to inspect the contents of IPv4 packets, it needs to know how to inspect tunneled traffic for any tunneling method used by the organization. A corollary to this is that if an organization is using a firewall to prohibit IPv6 coming into or going out of its network, that firewall needs to recognize and block all forms of v6-to-v4 tunneling.

For firewalls that permit IPv6 use, traffic with invalid source or destination IPv6 addresses should always be blocked—this is similar to blocking traffic with invalid IPv4 addresses. Since much more effort has been spent on making lists of invalid IPv4 addresses than on IPv6 addresses, finding lists of invalid IPv6 addresses can be difficult. Also, IPv6 allows network administrators to allocate addresses in their assigned ranges in different ways. This means that in a particular address range assigned to an organization, there can literally be trillions of invalid IPv6 addresses and only a few that are valid. By necessity, listing which IPv6 addresses are invalid will have to be less fine-grained than listing invalid IPv4 addresses, and the firewall rules that use these lists will be less effective than their IPv4 counterparts.

4.1.3 TCP and UDP

TCP and UDP are used by applications. An application server typically listens at a fixed TCP or UDP port, while application clients typically use any of a wide range of ports—and as with other aspects of firewall rulesets, deny by default policies should be used for incoming TCP and UDP traffic. Less stringent policies are generally used for outgoing TCP and UDP traffic because most organizations permit their users to access a wide range of external applications located on millions of external hosts.

An applications traffic matrix can be used to record the details of which activities are permitted and denied. Table 4-1 provides an example of a portion of a matrix for traffic attempting to enter a DMZ from the outside (i.e., through the organization's external firewall).

Table 4-1. Firewall Application Traffic Ruleset Matrix

Service[15]	Source	Destination	Action
HTTP	Internet	Public Web Servers	Permit
HTTPS	Internet	Web Servers for Access to Webmail	Permit
DNS	Internet	External DNS Server	Permit
Mail	Internet	External Mail Server	Permit
Network Time Protocol (NTP)	Internet	NTP Server	Permit
VPN Tunnels	Branch Offices	VPN Server	Permit
All Other Traffic	All Sources	Any Destination	Deny

In addition to allowing and blocking UDP and TCP traffic, many firewalls are also able to report or block malformed UDP and TCP traffic directed towards the firewall or to hosts protected by the firewall. This traffic is frequently used to scan for hosts, and may also be used in certain types of attacks. The firewall can help block such activity—or at least report when such activity is happening.

4.1.4 ICMP

Attackers can use various ICMP types and codes to perform reconnaissance or manipulate the flow of network traffic.[16] However, ICMP is needed for many useful things, such as getting reasonable performance across the Internet. Some firewall policies block all ICMP traffic, but this often leads to problems with diagnostics and performance. Other common policies allow all outgoing ICMP traffic, but limit incoming ICMP to those types and codes needed for path Maximum Transmission Unit (MTU) discovery and destination reachability.

To prevent malicious activity, firewalls at the network perimeter should deny all incoming and outgoing ICMP traffic except for those types and codes specifically permitted by the organization. For ICMP in IPv4, ICMP type 3 messages ("destination unreachable") should not be filtered because they are used for important network diagnostics. For ICMP in IPv6, many types of messages must be allowed in specific circumstances to enable various IPv6 features. See RFC 4890, *Recommendations for Filtering ICMPv6 Messages in Firewalls*, for detailed information on selecting which ICMPv6 types to allow or disallow for a particular firewall type.

4.1.5 IPsec Protocols

ESP and AH protocols are used for IPsec VPNs, and a firewall that blocks these protocols will not allow IPsec VPNs to pass. While blocking ESP can hinder the use of encryption to protect sensitive data, it can

[15] The amount of detail needed for the service varies among firewalls. For example, some firewalls have default protocol and port values for common services, such as TCP port 80 for HTTP. Other firewalls require the specific protocols (e.g., TCP, UDP) and port numbers (e.g., 80) to be entered by the administrator.

[16] ICMP type and code numbers are defined at http://www.iana.org/assignments/icmp-parameters.

also force users who would normally encrypt their data with ESP to allow it to be inspected—for example, by a stateful inspection firewall or an application-layer gateway.

Organizations should block ESP and AH except to and from specific addresses on the internal network—those addresses belong to IPsec gateways that are allowed to be VPN endpoints. Enforcing this policy will require people inside the organization to obtain the appropriate policy approval to open ESP and/or AH access to their IPsec routers. This will also reduce the amount of encrypted traffic coming from inside the network that cannot be examined by network security controls.

4.2 Policies Based on Applications

Most early firewall work involved simply blocking unwanted or suspicious traffic at the network boundary. Inbound application proxies take a different approach—they let traffic destined for a particular server into the network, but capture that traffic in a server that processes it like a port-based firewall. The application proxy approach provides an additional layer of security for incoming traffic by validating some of the traffic before it reaches the desired server. The theory is that the inbound application proxy's additional security layer can protect the server better than the server can protect itself—and can also remove malicious traffic before it reaches the server to help reduce server load. In some cases, an application proxy can remove traffic that the server might not be able to remove on its own because it has greater filtering capabilities. An application proxy also prevents the server from having direct access to the outside network.

If possible, inbound application proxies should be used in front of any server that does not have sufficient security features to protect it from application-specific attacks. The main considerations when deciding whether or not to use an inbound application proxy are:

- Is a suitable application proxy available?
- Is the server already sufficiently protected by existing firewalls?
- Can the main server remove malicious content as effectively as the application proxy?
- Is the latency caused by the proxy acceptable for the application?
- How easy it is to update the filtering rules on the main server and the application proxy to handle newly developed threats?

Unless an application proxy is significantly more robust than the server and easy to keep updated, it is usually best to stay with the application server alone. However, it is also important to consider the server's resources—if the server does not have sufficient resources to withstand attacks, the proxy could be used as a shield.

When an inbound application proxy is behind a firewall or in the firewall's DMZ, the firewall should be blocking based on IP addresses, as described earlier in this section, to reduce the load on the application proxy. Doing this puts more of the address-specific policy in a single place—the main firewall—and reduces the amount of traffic seen by the application proxy, freeing more power to filter content.

Outbound application proxies are useful for detecting systems that are making inappropriate or dangerous connections from inside the protected network. By far the most common type of outbound proxy is for HTTP. Outbound HTTP proxies allow an organization to filter dangerous content before it reaches the requesting PC. When an HTTP proxy filters content, it can alert the Web user that the site being visited sent the filtered content. The most prominent non-security benefit of HTTP proxies is caching Web pages for increased speed and decreased bandwidth use. Most organizations should employ HTTP proxies.

5. Firewall Planning and Implementation

This section focuses on the planning and implementation of firewalls in the enterprise. As with any new technology deployment, firewall planning and implementation should be addressed in a phased approach. A successful firewall deployment can be achieved by following a clear, step-by-step planning and implementation process. The use of a phased approach for deployment can minimize unforeseen issues and identify potential pitfalls early on. This section explores in depth each of the firewall planning and implementation phases, including:

1. **Plan.** The first phase of the process involves identifying all requirements that an organization should consider when determining which firewall to implement.

2. **Configure**. The second phase involves all facets of configuring the firewall platform. This includes installing hardware and software as well as setting up rules for the system.

3. **Test.** The next phase involves implementing and testing a prototype of the designed solution in a lab or test environment. The primary goals of testing are to evaluate the functionality, performance, scalability, and security of the solution, and to identify any issues—such as interoperability—with components.

4. **Deploy**. Once testing is completed and all issues are resolved, the next phase focuses on deployment of the firewall into the enterprise.

5. **Manage**. After the firewall has been deployed, it is managed throughout its lifecycle to include component maintenance and support for operational issues. This lifecycle process is repeated when enhancements or significant changes need to be incorporated into the solution.

5.1 Plan

The planning phase for choosing and implementing a firewall can begin only after an organization has determined that a firewall is needed to enforce the organization's security policy. This typically occurs following a risk assessment of the overall system. A risk assessment includes (1) the identification of threats and vulnerabilities in the information system; (2) the potential impact or magnitude of harm that a loss of confidentiality, integrity, or availability would have on the organization's assets or operations (including mission, function, image, or reputation) in the event of a threat exploitation of identified vulnerabilities; and (3) the identification and analysis of security controls for the information system[17].

Basic principles used in the planning of firewall deployments include:

- **Use devices as they were intended to be used.** Firewalls should not be constructed of equipment not meant for firewall use. For example, routers are meant to handle routing, not highly complex filtering, which can cause an excess burden on the router's processor. Additionally, firewalls should not be expected to provide other services, such as acting as a Web server or email server.

- **Create defense-in-depth.** Defense-in-depth involves creating multiple layers of security. This allows risk to be better managed, because if one layer of defense becomes compromised, another layer is there to contain the attack. In the case of firewalls, defense-in-depth can be accomplished by using multiple firewalls throughout an organization, including at the perimeter, in front of sensitive internal departments, and on individual computers. For defense-in-depth to be truly effective,

[17] For additional information about risk assessments, see NIST SP 800-30, *Risk Management Guide for Information Technology Systems* (http://csrc.nist.gov/publications/nistpubs/).

firewalls should be part of an overall security program that also includes products such as antivirus and intrusion detection software.

- **Pay attention to internal threats.** Focusing attention solely on external threats leaves the network wide open to attacks from within. These threats may not come directly from insiders, but can involve internal hosts infected by malware or otherwise compromised by external attackers. Important internal systems should be placed behind internal firewalls or DMZ environments.

Keep in mind that the expression "all rules are meant to be broken" applies when building firewalls. While firewall implementers should keep the above rules in mind during planning, every network and organization has unique requirements and idiosyncrasies that could require unique solutions.

Consider the following when purchasing and implementing a firewall solution:

- Security Capabilities
  - Which areas of the organization need to be protected (the perimeter, internal departments, remote office, individual hosts, specific services, mobile clients, etc.)?
  - Which types of firewall technologies will best address the kinds of traffic that need to be protected (packet filtering, stateful inspection, stateful protocol analysis, application-proxy/circuit-proxy gateway, etc.)?
  - What additional security features—such as intrusion detection capabilities, VPNs, and content filtering—does the firewall need to support?
- Performance (generally for network firewalls only)
  - What amount of throughput, maximum simultaneous connections, connections per second, and latency requirements must be met to prevent the firewall from being a bottleneck for network access, for both current and future traffic needs?
  - Are load balancing and failover functionally required to ensure high availability?
  - Is hardware-based vs. software-based firewall preference a consideration?
- Integration
  - Will the firewall require specific hardware to properly integrate within the organization's network infrastructure (specific power capabilities, specific type of network interface card [NIC], specific backup device, etc.)?
  - Does the firewall need to be compatible with other devices on the network that provide security or other services?
  - Will installing a firewall require changes to other areas of the network?
- Physical Environment (generally a consideration for network firewalls, although it may also apply to the centralized components of host-based or personal firewall implementations)
  - Where will the firewall be physically located to ensure physical security and protection from disasters?
  - Is there adequate shelf or rack space at the physical location where the firewall will be placed?

  - Will additional power, backup power, air conditioning, and/or network connections be required at the physical location?

- Personnel
  - Who will be responsible for managing the firewall?
  - Will system administrators require training before the firewall is deployed?
- Future Needs
  - Will the firewall meet the future needs of the organization (plans to move to IPv6, anticipated bandwidth requirements, compliance with regulations expected to be implemented, etc.)?

Other items to consider when purchasing and implementing host-based and personal firewalls include:

- Do workstations or servers meet the minimum system requirements of the firewall being evaluated?
- Will the firewall be compatible with other security software on the workstation or server (e.g., antivirus software)?
- Can the firewall be centrally managed and allow policies that enforce the organization's security policy to be pushed to users?
- Can the firewall report policy violations to a central server?
- Can the firewall be locked down to prevent anyone but administrators from modifying its settings?

5.2 Configure

The configuration phase involves all facets of configuring the firewall platform. This includes installing hardware and software, setting up rulesets, and configuring logging and alerting.

5.2.1 Hardware and Software

Once the firewall has been chosen and acquired, the hardware, operating system, and underlying firewall software should be installed for a software-based firewall. Next, for both software-based and hardware-based firewalls, patches and vendor updates should be installed on the system. During this stage, the firewall should also be hardened to decrease the risk of vulnerabilities and protect the system against unauthorized access. Any console software needed for remote access should also be installed at this time.

Network firewalls should be placed in a room that meets the product's recommended environmental requirements (temperature, humidity, space, power, etc.) and provides appropriate spacing between devices. This room should also be physically secured to prevent unauthorized personnel from accessing the firewall.

5.2.2 Configure Rulesets

Once hardware and software has been installed and secured, administrators can create the firewall's rulesets. These rulesets should implement the organization's firewall policy as described in Section 4, and should be as specific as possible with regards to the network traffic they control. To create a ruleset, it should first be determined what types of traffic (protocols, source and destination addresses, etc.) are required by approved applications for the organization. This should include protocols that the firewall itself may need (DNS, Simple Network Management Protocol [SNMP], logging, etc.)

The details of creating a ruleset vary by type of firewall and specific products. For example, many firewalls check traffic against rules in a sequential manner until a match is found. For these firewalls, rules with the highest chance of matching traffic patterns should be placed at the top of the list to improve firewall performance. Other firewalls have more complex ways of processing rulesets, such as first checking "deny" rules and then checking "allow" rules. Administrators should consult the firewall's documentation for recommendations on organizing rulesets to optimize performance and preventing ruleset errors that might create "holes" that allow unauthorized or unwanted traffic.

At minimum, the following rules should be defined:

- Port filtering should be enabled at the outer edge of the network, and probably at places inside the network as well.
- Content filtering should be done as close to the content receiver as possible.

Note that configuring rulesets is not an exact science, or even a craft. Many ways exist to define rules, and each organization will have its own needs and specific sets of personnel who should be involved in ruleset configuration.

5.2.3 Configure Logging and Alerts

The next step in the configuration process is to set up logging and alerts. Logging is a critical step in preventing and recovering from failures as well as ensuring that proper security configurations are set on the firewall. Proper logging can also provide vital information for responding to security incidents. The firewall should be configured to store logs locally—and, if possible, send them to a centralized log management infrastructure.

Real-time alerts should also be set up to notify administrators when important events occur on the firewall. Notifications may include the following:

- Any modifications or disabling of the firewall rules
- System reboots, disk shortages, and other operational events
- Secondary system status changes, if applicable.

5.3 Test

New firewalls should be tested and evaluated before deployment to ensure that they are working properly. Testing should be completed on a test network without connectivity to the production network. This test network should attempt to replicate the production network as faithfully as possible, including the network topology and network traffic that would travel through the firewall. Aspects of the solution to evaluate include the following:

- **Connectivity.** Users can establish and maintain connections through the firewall, and traffic that is specifically allowed by the security policy is permitted.
- **Blocking.** All traffic that is not allowed by the security policy is blocked.
- **Application Compatibility.** Host-based or personal firewall solutions do not break or interfere with the use of existing software applications. This includes network communications between application components.

- **Management.** Administrators can configure and manage the solution effectively and securely.
- **Logging.** Logging and data management function in accordance with the organization's policies and strategies.
- **Performance.** Solutions provide adequate performance during normal and peak usage. In many cases, the best way to test performance under the load of a prototype implementation is to use simulated traffic generators on a live test network to mimic the actual characteristics of expected traffic as closely as possible. Testing should incorporate a variety of applications that will traverse the firewall, especially those that are most likely to be affected by network throughput or latency issues.
- **Security of the Implementation.** The firewall implementation itself may contain vulnerabilities and weaknesses that attackers could exploit. Organizations with high security needs may want to perform vulnerability assessments against firewall components.
- **Component Interoperability.** Components of the firewall solution must function together properly. This is of greatest concern when a variety of components from different vendors are used.
- **Additional Features.** Additional features that will be used by the firewall—such as VPN and antivirus capabilities—should be tested to ensure they are working properly.

5.4 Deploy

Once testing is complete and all issues have been resolved, the next phase of the firewall planning and implementation model is deployment, which should be done in accordance with organization policies. Before deploying the firewall, administrators should notify users or owners of potentially affected systems of the planned deployment, and instruct them who to notify if they encounter any problems. Any changes required to other equipment, such as changing default routes, should also be coordinated as part of the firewall deployment.

If multiple firewalls are being deployed, including personal firewalls or at multiple branch offices, a gradual or phased approach should be considered by the organization. This will provide administrators with an opportunity to evaluate the impact of the firewall solution and resolve issues prior to enterprise-wide deployment.

5.5 Manage

This last phase of the firewall planning and implementation model is the longest lasting, because managing the solution involves maintaining firewall architecture, policies, software, and other components of the solution chosen to be deployed. One example of a typical maintenance action is testing and applying patches to firewall devices. [18] Policy rules may need to be updated as requirements change, such as when new applications or hosts are implemented within the network, and should also be reviewed periodically to ensure they remain in compliance with security policy. It is also important to monitor the performance of firewall components to ensure that potential resource issues are identified and addressed before components become overwhelmed. Logs and alerts should also be monitored continuously to identify threats—successful and unsuccessful—that are made to the system. Another important task is to perform periodic testing to verify that firewall rules are functioning as expected.

[18] For additional information about patch management, see NIST SP 800-40 Version 2, *Creating a Patch and Vulnerability Management Program* (http://csrc.nist.gov/publications/nistpubs/).

Changes to firewall rulesets or policies impact network security and should be managed by a formal process. Many firewalls have auditing of changes as part of their administrative interfaces, but this does not necessarily track policy changes. At a minimum, a log should be kept of all policy decisions and ruleset changes—and this log should somehow be associated with the firewall. For example, the log can be attached to the device physically, or the log file can be kept in the same part of the organization's inventory management system as the firewall.

Be aware that firewall rulesets can become increasingly complicated with age. For example, a new firewall ruleset might contain entries to accommodate only outbound user traffic and inbound email traffic (along with allowing the return inbound connections required by TCP/IP)—but will likely contain far more rules by the time the firewall system reaches the end of its first year in production. While new user or business requirements typically drive these changes, they can also reflect other influences within an organization.

Organizations may want to consider penetration testing to assess the overall security of their network environment. This testing can be used to verify that a firewall ruleset is performing as intended by generating network traffic and monitoring how it is handled by the firewall in comparison with its expected response. Penetration testing should be employed in addition to, rather than instead of, a conventional audit program.[19]

5.6 General Recommendations

The following recommendations for firewall planning and implementation will help administrators plan for firewall placement and implement their firewall policies.

- Placement and Deployment
  - Place a packet-filtering firewall at the edge of each discrete network in the organization.
  - Deploy firewalls at internal nodes in a network where one or more subnetworks have special needs that cannot be handled by the edge firewall.
  - Use a firewall with VPN features when confidentiality of traffic between two points on the network is needed.
  - Deploy personal firewalls on user systems that need protection beyond what can be provided by firewall(s) closer to the edge of the network.
- Firewall Policy
  - Deploy personal firewalls on all portable computers that may be used outside of a trusted organizational network.
  - Plan for the policy development and configuration of each firewall before it is deployed on the network.
  - Coordinate the policies of all firewalls in a network, and perform a regular review of all policies to ensure that organizational security policy is being met.

[19] For more information on penetration testing, see NIST SP 800-115 (Draft), *Technical Guide to Information Security Testing* (http://csrc.nist.gov/publications/nistpubs/).

- Only permit appropriate source and destination IP addresses in the traffic that flows in either direction through a firewall.
- Restrict the types of applications that can be reached from outside the protected network by blocking all non-approved TCP and UDP ports.
- Allow ICMP type 3 messages to pass through firewalls.

Appendix A—Glossary

Selected terms used in the publication are defined below.

Application-Proxy Gateway Firewall: An advanced firewall that combines lower layer access control with upper layer functionality, and includes a proxy agent that acts as an intermediary between two hosts that wish to communicate with each other.

Boundary Router: A router located at the organization's boundary with an untrusted external network. In the context of this document, a boundary router is configured to be a packet filter firewall.

Circuit-Level Gateway: A form of proxy that validates each connection before it is established, in much the same manner as stateful inspection.

Dedicated Proxy Server: A form of proxy that does not have firewalling capabilities.

Demilitarized Zone (DMZ): An interface on a routing firewall leading to a protected network that is different from the main network protected by the firewall. Traffic bound for the DMZ still goes through the firewall, and can have the firewall's protection policies applied.

Deny by Default: To block all inbound and outbound traffic that has not been expressly permitted by firewall policy.

Distributed Firewalling: Moving firewall capabilities from the network perimeter to device endpoints, such as placing a firewall in or directly in front of every endpoint and other appropriate devices in the network.

Egress Filtering: Filtering of outgoing network traffic.

Firewall: A device or program that controls the flow of network traffic between networks or hosts employing differing security postures.

Firewall Platform: The system device upon which a firewall is implemented. An example is a commercial operating system running on a personal computer.

Host-Based Firewall: A software-based firewall installed on a server to monitor and control its incoming and outgoing network traffic.

Ingress Filtering: Filtering of incoming network traffic.

Intranet: A network that employs services, applications, and protocols similar to those present in an Internet implementation, but without involving external connectivity. This allows data to be shared within the organization without private information being made available to individuals outside the intranet.

Malware: A program that is inserted into a system, usually covertly, with the intent of compromising the confidentiality, integrity, or availability of a victim's data, applications, or operating system, or otherwise annoying or disrupting the victim.

Network Address Translation (NAT): Used to hide internal system addresses from an external network through use of an addressing schema.

Packet Filter Firewall: A routing device that includes access control functionality for host addresses and communication sessions.

Personal Firewall: A software-based firewall installed on a desktop or laptop computer to monitor and control its incoming and outgoing network traffic.

Personal Firewall Appliance: A device that performs functions similar to a personal firewall for a group of computers on a home network.

Proxy Agent: A software application running on a firewall or on a dedicated proxy server that is capable of filtering a protocol and routing it between the interfaces of the device.

Ruleset: A set of directives that govern the access control functionality of a firewall. The firewall uses these directives to determine how packets should be routed between its interfaces.

Stateful Inspection Firewall: A firewall that can filter packets, track the state of connections, and block packets that deviate from the expected state.

Stateful Protocol Analysis Firewall: A stateful inspection firewall that includes an inspection engine able to analyze protocols at the network, transport, and application layers.

Appendix B—Acronyms and Abbreviations

Selected acronyms and abbreviations used in the publication are defined below.

AH Authentication Header
ALG Application Layer Gateways

CIDR Classless Interdomain Routing

DMZ Demilitarized Zone
DNS Domain Name System
DoS Denial of Service

ESP Encapsulating Security Payload

FISMA Federal Information Security Management Act
FTP File Transfer Protocol

HTTP Hypertext Transfer Protocol

IANA Internet Assigned Numbers Authority
ICMP Internet Control Message Protocol
IDPS Intrusion Detection and Prevention System
IDS Intrusion Detection System
IGMP Internet Group Management Protocol
IM Instant Messaging
IP Internet Protocol
IPsec Internet Protocol Security
IPv4 Internet Protocol version 4
IPv6 Internet Protocol version 6
ISP Internet Service Provider
IT Information Technology
ITL Information Technology Laboratory

LAN Local Area Network

MAC Media Access Control
MTU Maximum Transmission Unit

NAT Network Address Translation
NIC Network Interface Card
NIST National Institute of Standards and Technology
NTP Network Time Protocol

OMB Office of Management and Budget

PBX Private Branch Exchange
PC Personal Computer

RADIUS Remote Authentication Dial In User Service
RFC Request for Comment

SMTP	Simple Mail Transfer Protocol
SNMP	Simple Network Management Protocol
SP	Special Publication
SSL	Secure Sockets Layer
TCP	Transmission Control Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
TLS	Transport Layer Security
UDP	User Datagram Protocol
URL	Uniform Resource Locator
VPN	Virtual Private Network
VPNC	Virtual Private Network Consortium
WAN	Wide Area Network
XML	Extensible Markup Language

Appendix C—Resources

The lists below provide examples of resources that may be helpful.

Print Resources

Allen, Julia H, *The CERT Guide to System and Network Security Practices*, Addison-Wesley, 2001

Chapman, Brent, et al, *Building Internet Firewalls, 2nd Edition*, O'Reilly Media, Inc., 2000.

Cheswick, William R., et al, *Firewalls and Internet Security: Repelling the Wily Hacker*, 2nd Edition, 2003.

Deal, Richard A, *Cisco Router Firewall Security*, Cisco Press, 2005.

Noonan, Wes and Dubrawsky, Ido, *Firewall Fundamentals*, Cisco Press, 2006.

Northcutt, Stephen, et al, *Inside Network Perimeter Security, 2nd Edition*, Sams, 2005.

Rash, Michael, *Linux Firewalls: Attack Detection and Response with iptables, psad, and fwsnort*, No Starch Press, 2007.

Shimonski, Robert J., et al, *Building DMZs for Enterprise Networks*, Syngress Publishing, Inc., 2003.

Shinder, Thomas, W., et al, *The Best Damn Firewall Book Period, Second Edition*, Syngress Publishing, Inc., 2007.

NIST Documents and Resource Sites

Resource Name	Uniform Resource Locator (URL)
NIST National Checklist Program	http://checklists.nist.gov/
NIST SP 800-18 Revision 1, *Guide for Developing Security Plans for Federal Information Systems*	http://csrc.nist.gov/publications/nistpubs/800-18-Rev1/sp800-18-Rev1-final.pdf
NIST SP 800-30, *Risk Management Guide for Information Technology Systems*	http://csrc.nist.gov/publications/nistpubs/800-30/sp800-30.pdf
NIST SP 800-40 Version 2, *Creating a Patch and Vulnerability Management Program*	http://csrc.nist.gov/publications/nistpubs/800-40-Ver2/SP800-40v2.pdf
NIST SP 800-44 Version 2, *Guidelines on Securing Public Web Servers*	http://csrc.nist.gov/publications/nistpubs/800-44-ver2/SP800-44v2.pdf
NIST SP 800-45 Version 2, *Guidelines on Electronic Mail Security*	http://csrc.nist.gov/publications/nistpubs/800-45-version2/SP800-45v2.pdf
NIST SP 800-46, *Security for Telecommuting and Broadband Communications*	http://csrc.nist.gov/publications/nistpubs/800-46/sp800-46.pdf
NIST SP 800-52, *Guidelines for the Selection and Use of Transport Layer Security (TLS) Implementations*	http://csrc.nist.gov/publications/nistpubs/800-52/SP800-52.pdf
NIST SP 800-61 Revision 1, *Computer Security Incident Handling Guide*	http://csrc.nist.gov/publications/nistpubs/800-61-rev1/SP800-61rev1.pdf
NIST SP 800-70, *Security Configuration Checklists Program for IT Products – Guidance for Checklists Users and Developers*	http://csrc.nist.gov/checklists/docs/SP_800-70_20050526.pdf

Resource Name	Uniform Resource Locator (URL)
NIST SP 800-77, *Guide to IPsec VPNs*	http://csrc.nist.gov/publications/nistpubs/800-77/sp800-77.pdf
NIST SP 800-81, *Secure Domain Name System (DNS) Deployment Guide*	http://csrc.nist.gov/publications/nistpubs/800-81/SP800-81.pdf
NIST SP 800-86, *Guide to Integrating Forensic Techniques into Incident Response*	http://csrc.nist.gov/publications/nistpubs/800-86/SP800-86.pdf
NIST SP 800-92, *Guide to Computer Security Log Management*	http://csrc.nist.gov/publications/nistpubs/800-92/SP800-92.pdf
NIST SP 800-94, *Guide to Intrusion Detection and Prevention Systems (IDPS)*	http://csrc.nist.gov/publications/nistpubs/800-94/SP800-94.pdf
NIST SP 800-95, *Guide to Secure Web Services*	http://csrc.nist.gov/publications/nistpubs/800-95/SP800-95.pdf
NIST SP 800-97, *Establishing Wireless Robust Security Networks: A Guide to IEEE 802.11i*	http://csrc.nist.gov/publications/nistpubs/800-97/SP800-97.pdf
NIST SP 800-113, *Guide to SSL VPNs*	http://csrc.nist.gov/publications/nistpubs/800-113/SP800-113.pdf
NIST SP 800-115 (Draft), *Technical Guide to Information Security Testing*	http://csrc.nist.gov/publications/PubsSPs.html

Other Technical Resource Sites and Documents

Resource Name	Uniform Resource Locator (URL)
Achieving Defense-in-Depth with Internal Firewalls	http://www.sans.org/reading_room/whitepapers/firewalls/797.php?portal=a4d358dbd051422110d917753a0ebb7c
An Introduction to Network Firewalls and the Firewall Selection Process	http://www.more.net/technical/netserv/tcpip/firewalls/
Best Practices for Managing Firewall Logs	http://www.zdnet.com.au/insight/print.htm?TYPE=story&AT=120265680-139023731t-110000100c
Comparison of Firewall, Intrusion Prevention, and Antivirus Technologies	http://www.juniper.net/solutions/literature/white_papers/200063.pdf
Defense in Depth: Foundations for Secure and Resilient IT Enterprises	http://www.cert.org/archive/pdf/Defense_in_Depth092106.pdf
Firewall Evolution – Deep Packet Inspection	http://www.securityfocus.com/infocus/1716
Handbook of Firewall Architectures	http://www.securecomputing.com/pdf/SIDE--Ch-ExamFirewallTP.pdf
How do Circuit-Level Gateways and Application-Level Gateways Differ?	http://searchsecurity.techtarget.com/expert/KnowledgebaseAnswer/0,289625,sid14_gci1197999,00.html
IPv6 Transition Guidance	http://www.cio.gov/documents/IPv6_Transition_Guidance.doc
National Vulnerability Database	http://nvd.nist.gov/
The Perils of Deep Packet Inspection	http://www.securityfocus.com/infocus/1817
RFC Editor Homepage	http://www.rfc-editor.org/
Security Implications of IPv6	http://www.iss.net/documents/whitepapers/IPv6.pdf
Transparent, Bridging Firewall Devices	http://www.securityfocus.com/infocus/1737
Trusted Computing Group: Trusted Network Connect	https://www.trustedcomputinggroup.org/groups/network/
The Web Services Advisor: XML Firewalls	http://searchwebservices.techtarget.com/tip/1,289483,sid26_gci855052,00.html