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

Special Publication 800-41

Guidelines on Firewalls and Firewall Policy

Recommendations of the National Institute of Standards and Technology

John Wack, Ken Cutler, Jamie Pole

NIST Special Publication 800-41

Guidelines on Firewalls and Firewall Policy

Recommendations of the National Institute of Standards and Technology

John Wack, Ken Cutler*, Jamie Pole*

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

*MIS Training Institute

January 2002

U.S. Department of Commerce
Donald L. Evans, Secretary

Technology Administration
Phillip J. Bond, Under Secretary for Technology

National Institute of Standards and Technology
Arden L. Bement, Jr., 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
Natl. Inst. Stand. Technol. Spec. Publ. 800-41, 75 pages (Jan. 2002)

U.S. GOVERNMENT PRINTING OFFICE
WASHINGTON: 2001

For sale by the Superintendent of Documents, U.S. Government Printing Office
Internet: bookstore.gpo.gov — Phone: (202) 512-1800 — Fax: (202) 512-2250
Mail: Stop SSOP, Washington, DC 20402-0001

Foreword

This document provides guidelines for Federal organizations' acquisition and use of security-related Information Technology (IT) products. These guidelines provide advice to agencies for sensitive (i.e., non-national security) unclassified systems. NIST's advice is given in the context of larger recommendations regarding computer systems security.

NIST developed this document in furtherance of its statutory responsibilities under the Computer Security Act of 1987 and the Information Technology Management Reform Active of 1996 (specifically section 15 of the United States Code (U.S.C.) 278 g-3(a)(5)). This is not a guideline within the meaning of 15 U.S.C. 278 g-3 (a)(3).

These guidelines are for use by Federal organizations that process sensitive information[1]. They are consistent with the requirements of OMB Circular A-130, Appendix III.

The guidelines herein are not mandatory and binding standards. This document may be used voluntarily by non-governmental organizations. It is not subject to copyright.

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

Acknowledgements

The authors wish to express their thanks to staff at NIST and at other organizations who reviewed drafts of this document. In particular, 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, provided valuable insights that contributed substantially to the technical content of this document.

[1] The Computer Security Act provides a broad definition of the term "sensitive information," namely "any information, the loss, misuse, or unauthorized access to or modification of which could adversely affect the national interest or the conduct of federal programs, or the privacy to which individuals are entitled under section 552a of title 5, United States Code (the Privacy Act), but which has not been specifically authorized under criteria established by an Executive Order or an Act of Congress to be kept secret in the interest of national defense or foreign policy."

Table of Contents

List of Figures

List of Tables

Executive Summary

Firewall technology has matured to the extent that today's firewalls can coordinate security with other firewalls and intrusion detection systems. They can scan for viruses and malicious code in electronic mail and web pages. Firewalls are now standard equipment for Internet connections. Home users who connect to commercial Internet service providers via dial-up or via cable/DSL are also using personal firewalls and firewall appliances to secure their connections.

Firewalls protect sites from exploitation of inherent vulnerabilities in the TCP/IP protocol suite. Additionally, they help mitigate security problems associated with insecure systems and the problems inherent in providing robust system security for large numbers of computers. There are several types of firewalls, ranging from boundary routers that can provide access control on Internet Protocol packets, to more powerful firewalls that can close more vulnerabilities in the TCP/IP protocol suite, to even more powerful firewalls that can filter on the content of the traffic.

The type of firewall to use depends on several factors, including the size of the site, the amount of traffic, the sensitivity of systems and data, and the applications required by the organization. The choice of firewall should largely be driven by its feature set, rather than the type of firewall, however. A standard firewall configuration involves using a router with access control capability at the boundary of the organization's network, and then using a more powerful firewall located behind the router.

Firewall environments are made up of firewall devices and associated systems and applications designed to work together. For example, one site may use a firewall environment composed of a boundary router, a main firewall, and intrusion detection systems connected to the protected network and the network between the router and main firewall. To provide secure remote access, the firewall may incorporate a virtual private network (VPN) server to encrypt traffic between the firewall and telecommuters or between the firewall and other sites on the Internet. The firewall environment may incorporate specialized networks for locating externally accessible servers such as for websites and email. The configuration of the firewall environment must be done carefully so as to minimize complexity and management, but at the same time provide adequate protection for the organization's networks. As always, a policy is essential.

Firewalls are vulnerable themselves to misconfigurations and failures to apply needed patches or other security enhancements. Accordingly, firewall configuration and administration must be performed carefully and organizations should also stay current on new vulnerabilities and incidents. While a firewall is an organization's first line of defense, organizations should practice a defense in depth strategy, in which layers of firewalls and other security systems are used throughout the network. Most importantly, organizations should strive to maintain all systems in a secure manner and not depend solely on the firewall to stop security threats. Organizations need backup plans in case the firewall fails.

This document contains numerous recommendations for choosing, configuring, and maintaining firewalls. These recommendations are summarized in Appendix C.

1. Introduction

Firewall technology has improved substantially since it was introduced in the early 1990s. The early firewall technology started with simple packet-filtering firewalls and progressed to more sophisticated firewalls capable of examining multiple layers of network activity and content. As the Internet has developed into the modern, complex network of today, Internet security has become more problematic, with break-ins and attacks now so commonplace as to be considered part of doing business. Now, firewall technology is a standard part of any organization's network security architecture. Today, home users on commercial dial-in and cable/DSL connections routinely employ personal firewalls and firewall appliances.

Modern firewalls are able to work in conjunction with tools such as intrusion detection monitors and email/web content scanners for viruses and harmful application code. But firewalls alone do not provide complete protection from Internet-borne problems. As a result, they are just one part of a total information security program. Generally firewalls are viewed as the first line of defense, however it may be better to view them as the *last* line of defense for an organization; organizations should still make the security of their internal systems a high priority. Internal servers, personal computers, and other systems should be kept up-to-date with security patches and anti-virus software.

1.1. Document Purpose and Scope

This document provides introductory information about firewalls and firewall policy primarily to assist those responsible for network security. It addresses concepts relating to the design, selection, deployment, and management of firewalls and firewall environments. This document is not intended to provide a mandatory framework for firewalls and firewall environments, but rather to present suggested approaches to the topic.

This document is an update to NIST Special Publication 800-10, *Keeping Your Site Comfortably Secure: An Introduction to Firewall Technology*.[2] That document dealt with the firewall landscape of 1994, and while the basic aspects of firewalls described in Special Publication 800-10 are still relevant, numerous aspects of firewall technology have changed.

Special Publication 800-10 dealt with the basics of Internet Protocol (IP) packet filtering and application gateway firewalls, and outlined basic firewall configurations and policy. This document covers IP filtering with more recent policy recommendations, and deals generally with hybrid firewalls that can filter packets and perform application gateway (proxy) services. This document also contains specific recommendations for policy as well as a simple methodology for creating firewall policy.

1.2. Audience and Assumptions

The intended audience is technical personnel, as well as management personnel who might require a technical basis for supporting a decision-making process. Non-technical manage-

[2] Available at http://csrc.nist.gov.

ment and those wishing to increase their knowledge of firewalls may find this document useful as well. This document assumes some knowledge of TCP/IP (Transmission Control Protocol/Internet Protocol), the protocol suite used by the Internet, as well as various other aspects of networking and information security. Less-technical readers may find Special Publication 800-10 a useful starting point for firewall concepts.

1.3. Document Organization

The remainder of this document is organized as follows:

Chapter 2 contains a review of the Open Systems Interconnect (OSI) protocol stack and uses this to describe a number of different firewall platforms, including packet filter firewalls, stateful firewalls, and application-proxy firewalls.

Chapter 3 describes various firewall environments, i.e., components that combined, constitute a firewall solution. It contains suggestions for positioning firewalls and enabling them to work in conjunction with other security tools. Chapter 3 also describes other aspects of modern firewalling such as Virtual Private Networks (VPNs), IP address translation, and filtering of content such as email attachments.

Chapters 4 and 5 contain detailed information useful for those who administer firewalls and configure firewall policy. Chapter 4 describes firewall policy, how it should fit within an overall policy framework, and then presents a suggested minimum policy that can be tailored to suit many environments. Chapter 5 presents suggestions for implementing and managing firewall administration.

Appendix A defines terminology used in this document. Appendix B contains resources and on-line links for more information about computer security in general and firewalls in particular. Appendix C summarizes recommendations contained in the main chapters and recommends additional firewall measures.

2. Overview of Firewall Platforms

The concept of network firewalls has been debated and discussed since the inception of secure connectivity requirements. This chapter contains an overview of firewall capabilities and then goes on to describe several types of firewalls in detail.

2.1. General Introduction to Firewall Technology

Network firewalls are devices or systems that control the flow of network traffic between networks employing differing security postures. In most modern applications, firewalls and firewall environments are discussed in the context of Internet connectivity and the TCP/IP protocol suite. However, firewalls have applicability in network environments that do not include or require Internet connectivity. For example, many corporate enterprise networks employ firewalls to restrict connectivity to and from internal networks servicing more sensitive functions, such as the accounting or personnel department. By employing firewalls to control connectivity to these areas, an organization can prevent unauthorized access to the respective systems and resources within the more sensitive areas. The inclusion of a proper firewall or firewall environment can therefore provide an additional layer of security that would not otherwise be available.

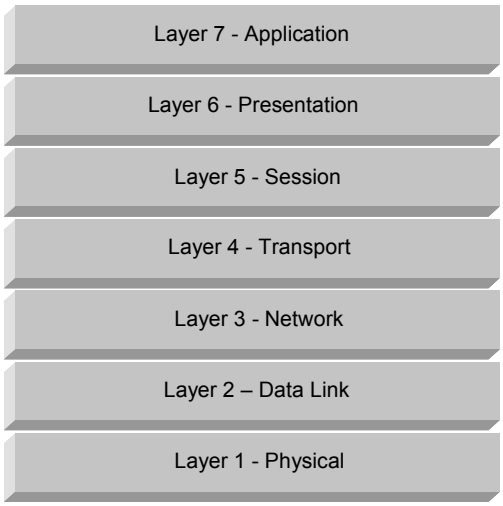

Figure 2.1: OSI Communications Stack

There are several types of firewall platforms currently available from vendors. One way of comparing the capabilities of the firewall platforms is by examining the aspects of the Open Systems Interconnect (OSI) model that each given firewall platform is able to function with and can make use of. The OSI model is an abstraction of network communications between computer systems and network devices. The exact details of the OSI model are outside the scope of this document, but those layers relevant to the firewall topic are addressed.

A graphic depiction of the OSI model in Figure 2.1 shows a stack of networking layers. The component layering illustrated is only for discussion purposes and not meant to imply

any structural relationship. As a brief summary, the OSI model exists mainly to simplify the process of understanding how computer systems communicate in a network. Layer 1 represents the actual physical communication hardware and media such as Ethernet. Layer 2 represents the layer at which network traffic delivery on Local Area Networks (LANs) occurs. Layer 2 is also the first layer that contains addressing that can identify a single specific machine. The addresses are assigned to network interfaces and are referred to as MAC, or Media Access Control addresses. An Ethernet address belonging to an Ethernet card is an example of a Layer 2 MAC address.

Layer 3 is the layer that accomplishes delivery of network traffic on Wide Area Networks (WANs). On the Internet, Layer 3 addresses are referred to as Internet Protocol (IP) addresses; the addresses are normally unique but in circumstances involving Network Address Translation (NAT), it is possible that multiple physical systems are represented by a single Layer 3 IP address. Layer 4 identifies specific network applications and communication *sessions* as opposed to network addresses; a system may have any number of Layer 4 sessions with other systems on the same network. Terminology associated with the TCP/IP protocol suite includes the notion of *ports*, which can be viewed as end points for sessions: a *source* port number identifies the communication session on the originating system; a *destination* port identifies the communication session of the destination system. The upper layers (5, 6, and 7) representing end-user applications and systems, are shown here for illustration purposes only.

For the purposes of this document, modern firewalls operate on the following OSI model layers as shown in Figure 2.2.

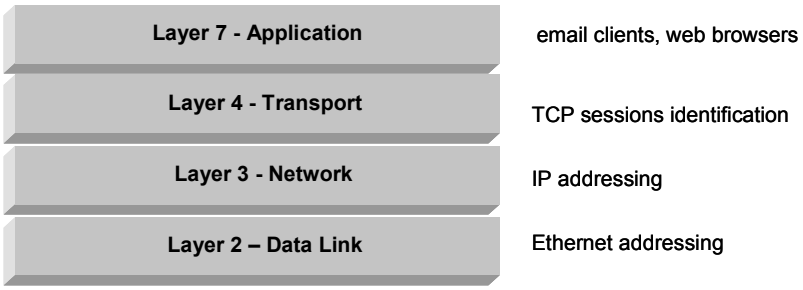

Figure 2.2: OSI Layers Operated on Modern Firewalls

Basic firewalls will operate on a smaller number of layers; more advanced firewalls will cover a larger number of layers. In terms of functionality, firewalls capable of examining a larger number of layers are more thorough and effective. Additional layer coverage also increases the configuration granularity present in the firewall; adding layer awareness allows the firewall to accommodate advanced applications and protocols. Increasing the layers a firewall can examine also allows the firewall to provide services that are very user-oriented, such as user authentication. A firewall that function with layers 2 and 3 only does not usually deal with specific users, but a higher end application-proxy gateway firewall can enforce user authentication as well as logging events to specific users.

Independent of firewall architecture, there can be many add-on services. Some of these services include Network Address Translation (NAT), Dynamic Host Configuration Protocol (DHCP), encryption functionality such as Virtual Private Networks (VPNs), and

application content filtering. These services are discussed in the balance of this section with the exception of NAT, which is discussed in Section 2.7.

New firewalls support the Dynamic Host Configuration Protocol (DHCP) to allocate IP addresses for those addresses (of systems) that will be subject to the firewall's security controls and to simplify network management. DHCP was originally a proprietary set of extensions to the original bootstrap protocol for network devices without resident operating systems (BOOTP). The DHCP specification is now supported on nearly all business and consumer operating systems and is widely used because it makes the network administration of IP addresses easier. A commonplace use for DHCP is for dial-in connections; often the dial-in server assigns a dynamically generated IP address to the dial-in user's system using DHCP.

Firewalls can also act as Virtual Private Network (VPN) gateways. Thus, an organization or agency can send unencrypted network traffic from systems behind the firewall to other remote systems behind a cooperating VPN gateway; the firewall encrypts the traffic and forwards it to the remote VPN gateway, which decrypts it and passes it on to the destination systems. Most of the more popular firewalls nowadays incorporate this type of functionality (VPNs are discussed in greater detail in Section 3.3).

The final add-on involves active content filtering technologies. This mechanism differs from the normal function of a firewall in that the firewall can also be capable of filtering the actual application data at layer 7 that seeks to traverse the firewall. For example, this mechanism might be employed to scan email attachments and remove viruses. It is also widely used to filter the more dangerous active web-enabling technologies, such as Java™[3], JavaScript, and ActiveX®[4][5]. Or, it can be used to filter on contents or keywords to restrict web access to inappropriate sites or domains. However, firewall-based content filtering should not be relied upon as the sole content filtering mechanism for an organization or agency; it is possible to bypass these filters through the use of compression or encryption or other techniques.

2.2. Packet Filter Firewalls

The most basic, fundamental type of firewall is called a packet filter. Packet filter firewalls are essentially routing devices that include access control functionality for system addresses and communication sessions. The access control functionality of a packet filter firewall is governed by a set of directives collectively referred to as a ruleset. A sample packet filter firewall ruleset is included at the end of this section in Table 2.1.

---

[3] Sun, Sun Microsystems, the Sun Logo, Solaris, Java, and Jini are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries.

[4] ActiveX, Windows, Windows NT, Windows 2000, Windows XP, and Word, are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

[5] See NIST ITL Bulletin *Security Implications of Active Content*, March 2000, and NIST Special Publication 800-28, *Guidelines for Active Content and Mobile Code*, at http://csrc.nist.gov.

In their most basic form, packet filters operate at Layer 3 (Network) of the OSI model. This basic functionality is designed to provide network access control based upon several pieces of information contained in a network packet:

- The *source address* of the packet, i.e., the Layer 3 address of the computer system or device the network packet originated from (an IP address such as 192.168.1.1).
- The *destination address* of the packet, i.e., the Layer 3 address of the computer system or device the network packet is trying to reach (e.g., 192.168.1.2).
- The *type of traffic*, that is, the specific network protocol being used to communicate between the source and destination systems or devices (often Ethernet at Layer 2 and IP at Layer 3).
- Possibly some *characteristics of the Layer 4 communications sessions*, such as the source and destination ports of the sessions (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).
- Sometimes, information pertaining to *which interface of the router the packet came from* and which interface of the router the packet is destined for; this is useful for routers with 3 or more network interfaces.

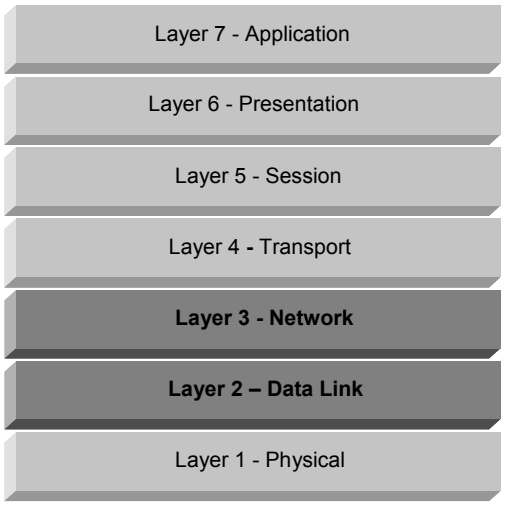

Figure 2.3: OSI Layers Addressed by Packet Filters

Packet filter firewalls are commonly deployed within TCP/IP network infrastructures; however, they can also be deployed in any network infrastructure that relies on Layer 3 addressing, including IPX (Novell NetWare) networks. In the context of modern network infrastructures, firewalling at Layer 2 is used in load balancing and/or high-availability applications in which two or more firewalls are employed to increase throughput or for fail-safe operations.

Packet filtering firewalls and routers can also filter network traffic based upon certain characteristics of that traffic, such as whether the packet's Layer 3 protocol might be the

Internet Control Message Protocol[6] (ICMP) – attackers have used this protocol to flood networks with traffic, thereby creating distributed denial-of-service (DDOS) attacks[7]. Packet filter firewalls also have the capability to block other attacks that take advantage of weaknesses in the TCP/IP suite.

Boundary Routers

Packet filter firewalls have two main strengths: speed and flexibility. Since packet filters do not usually examine data above Layer 3 of the OSI model, they can operate very quickly. Likewise, since most modern network protocols can be accommodated using Layer 3 and below, packet filter firewalls can be used to secure nearly any type of network communication or protocol. This simplicity allows packet filter firewalls to be deployed into nearly any enterprise network infrastructure. An important point is that their speed and flexibility, as well as capability to block denial-of-service and related attacks, makes them ideal for placement at the outermost boundary with an untrusted network. The packet filter, referred to as a *boundary router*, can block certain attacks, possibly filter unwanted protocols, perform simple access control, and then pass the traffic onto other firewalls that examine higher layers of the OSI stack.

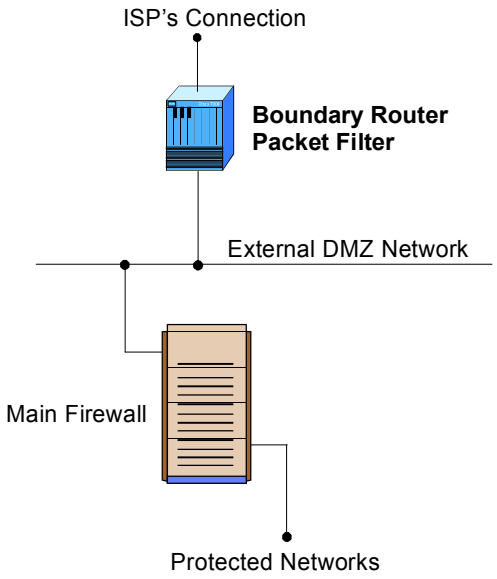

Figure 2.4: Packet Filter used as Boundary Router

Figure 2.4 shows a packet filter used as a boundary router. The router accepts packets from the untrusted network connection, which typically would be another router owned or controlled by the Internet Service Provider (ISP). The router then performs access control according to the policy in place, e.g., block SNMP (Simple Network Management Protocol), permit HTTP (Hypertext Transport Protocol), etc. It then passes the packets to other

[6] The ICMP protocol is at the same OSI layer as the IP protocol and is used primarily for determining routing paths.

[7] See NIST ITL Bulletins *Computer Attacks: What They Are and How to Defend Against Them*, May 1999, and *Mitigating Emerging Hacker Threats*, June, 2000, at http://csrc.nist.gov

more powerful firewalls for more access control and filtering operations at higher layers of the OSI stack. Figure 2.4 also shows an internal, less trusted network between the boundary router and an inner firewall, sometimes referred to as the external DMZ (Demilitarized Zone) network.

Basic Weaknesses Associated with Packet Filters

Packet filter firewalls also possess several weaknesses:

- Because packet filter firewalls do not examine upper-layer data, they cannot prevent attacks that employ application-specific vulnerabilities or functions. For example, a packet filter firewall cannot block specific application commands; if a packet filter firewall allows a given application, all functions available within that application will be permitted.
- Because of the limited information available to the firewall, the logging functionality present in packet filter firewalls is limited. Packet filter logs normally contain the same information used to make access control decisions (source address, destination address, and traffic type).
- Most packet filter firewalls do not support advanced user authentication schemes. Once again, this limitation is mostly due to the lack of upper-layer functionality by the firewall.
- They are generally vulnerable to attacks and exploits that take advantage of problems within the TCP/IP specification and protocol stack, such as *network layer address spoofing*. Many packet filter firewalls cannot detect a network packet in which the OSI Layer 3 addressing information has been altered. Spoofing attacks are generally employed by intruders to bypass the security controls implemented in a firewall platform.
- Finally, due to the small number of variables used in access control decisions, packet filter firewalls are susceptible to security breaches caused by improper configurations. In other words, it is easy to accidentally configure a packet filter firewall to allow traffic types, sources, and destinations that should be denied based upon an organization's information security policy.

Consequently, packet filter firewalls are very suitable for high-speed environments where logging and user authentication with network resources are not important.

Since current firewall technology includes many features and functionality, it is difficult to identify a single firewall that contains only packet filter features. The closest example would be a network router employing coded access control lists to handle network traffic. The simplicity of packet filter firewalls also easily facilitates the implementation of high-availability and hot failover[8] solutions; several vendors offer hardware and software solutions for both high-availability and hot failover. Most SOHO (Small Office Home Office) firewall appliances and default operating system firewalls are packet filter firewalls.

---

[8] Hot failover firewall systems incorporate at least one backup firewall. When the primary firewall is taken off line, the hot failover firewall comes on-line and maintains all existing communications sessions; no disruption of communications occurs.

	Source Address	Source Port	Destination Address	Destination Port	Action	Description
1	Any	Any	192.168.1.0	> 1023	Allow	Rule to allow return TCP Connections to internal subnet
2	192.168.1.1	Any	Any	Any	Deny	Prevent Firewall system itself from directly connecting to anything
3	Any	Any	192.168.1.1	Any	Deny	Prevent External users from directly accessing the Firewall system.
4	192.168.1.0	Any	Any	Any	Allow	Internal Users can access External servers
5	Any	Any	192.168.1.2	SMTP	Allow	Allow External Users to send email in
6	Any	Any	192.168.1.3	HTTP	Allow	Allow External Users to access WWW server
7	Any	Any	Any	Any	Deny	"Catch-All" Rule - Everything not previously allowed is explicitly denied

Table 2.1: Sample Packet Filter Firewall Ruleset

Packet Filter Rulesets

Table 2.1 shows a sample of a packet filter firewall ruleset for an imaginary network of IP address 192.168.1.0, with the "0" indicating that the network has addresses that range from 192.168.1.0 to 192.168.1.254. For most firewalls, the ruleset would be much larger and detailed. The firewall would normally accept a packet and examine its source and destination addresses and ports, and determine what protocol is in use. From there, the firewall would start at the top of the ruleset and work down through the rules. Whenever a rule that permits or denies the packet is found, one of the following actions is taken:

- *Accept*: the firewall passes the packet through the firewall as requested, subject to whatever logging capabilities may or may not be in place.
- *Deny*: the firewall drops the packet, without passing it through the firewall. Once the packet is dropped, an error message is returned to the source system. The "Deny" action may or may not generate log entries depending on the firewall's ruleset configuration.
- *Discard*: the firewall not only drops the packet, but it does not return an error message to the source system. This particular action is used to implement the "black hole" methodology in which a firewall does not reveal its presence to an outsider. As with the other actions, the "Discard" action may or may not generate log entries.

In Table 2.1, the first rule permits return packets from external systems to return to the internal systems, thus completing the connection (it is assumed that if a connection to an

external system was permitted, then the return packets from the external system should be permitted as well). The second rule prohibits the firewall from forwarding any packets with a source address from the firewall; this condition would indicate that an attacker is spoofing the firewall's address in the hopes that the firewall would pass this packet to an internal destination. As a result, the destination might then accept the packet since it would appear to have come from the trusted firewall. The third rule simply blocks external packets from directly accessing the firewall.

The fourth rule allows internal systems to connect to external systems, using any external addresses and any protocol. Rules 5 and 6 allow external packets past the firewall if they contain SMTP (Simple Mail Transport Protocol) data or HTTP data, that is, email and web data respectively. The final rule, a very important one, blocks any other packets from the outside. One can deduce, then, that the information security policy for the network is as follows:

- Any type of access from the inside to the outside is allowed.
- No access originating from the outside to the inside is allowed except for SMTP and HTTP.
- Also, the SMTP and HTTP servers are positioned "behind" the firewall.

An important point is that if the last rule were accidentally skipped, *all traffic originating from the outside would be permitted*. When the ruleset is much longer and more detailed, mistakes can be made that could prove disastrous. The ruleset should be examined very carefully before implementation, and regularly thereafter, not only to ensure that correct protocols are allowed based on business requirements, but also to minimize logical errors when new rules are added.

A final note about packet filters: filtering can occur on *outbound* as well as inbound traffic. An organization could choose to restrict the types of traffic originating from within the organization, such as blocking all outbound FTP traffic. In practice, outbound filtering is often employed on IP addresses and application traffic, for example, to block all users, internal and external, from connecting to certain systems such as the packet filter itself, backup servers, and other sensitive systems.

2.3. Stateful Inspection Firewalls

Stateful inspection firewalls are packet filters that incorporate added awareness of the OSI model data at Layer 4, as shown in Figure 2.5.

Stateful inspection evolved from the need to accommodate certain features of the TCP/IP protocol suite that make firewall deployment difficult. When a TCP (connection-oriented transport) application creates a session with a remote host system, a port is also created on the source system for the purpose of receiving network traffic from the destination system. According to the TCP specifications, this client *source port* will be some number greater than 1023 and less than 16384. According to convention, the destination port on the remote host will likely be a "low-numbered" port, less than 1024. This will be 25 for SMTP, for example.

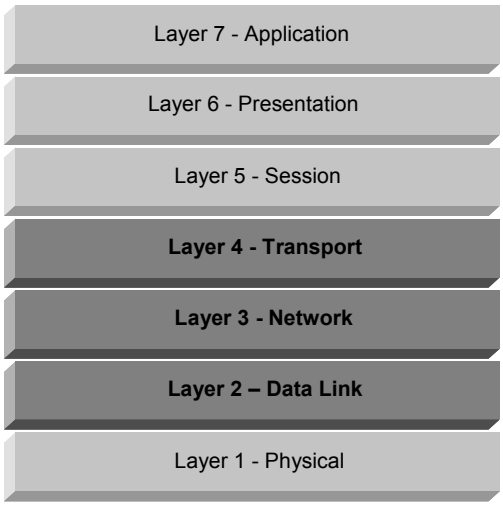

Figure 2.5: OSI Layers Addressed by Stateful Inspection

Packet filter firewalls must permit inbound network traffic on all of these "high-numbered" ports for connection-oriented transport to occur, i.e., return packets from the destination system. Opening this many ports creates an immense risk of intrusion by un-authorized users who may employ a variety of techniques to abuse the expected conventions.

Table 2.2 shows the first line of the packet filter ruleset from Table 2.1, which permits any inbound connection if the destination port is above 1023. Stateful inspection firewalls solve this problem by creating a directory of outbound TCP connections, along with each session's corresponding "high-numbered" client port. This "state table" is then used to validate any inbound traffic. The stateful inspection solution is more secure because the firewall tracks client ports individually rather than opening all "high-numbered" ports for external access.

	Source Address	Source Port	Destination Address	Destination Port	Action	Description
1	Any	Any	192.168.1.0	> 1023	Allow	Rule to allow return TCP Connections to internal subnet

Table 2.2: Return Connection Rule

In essence, stateful inspection firewalls add Layer 4 awareness to the standard packet filter architecture. Stateful inspection firewalls share the strengths and weaknesses of packet filter firewalls, but due to the state table implementation, stateful inspection firewalls are generally considered to be more secure than packet filter firewalls. Table 2.3 shows an example of a state table from a stateful packet filter firewall:

Source Address	Source Port	Destination Address	Destination Port	Connection State
192.168.1.100	1030	210.9.88.29	80	Established
192.168.1.102	1031	216.32.42.123	80	Established
192.168.1.101	1033	173.66.32.122	25	Established
192.168.1.106	1035	177.231.32.12	79	Established
223.43.21.231	1990	192.168.1.6	80	Established
219.22.123.32	2112	192.168.1.6	80	Established
210.99.212.18	3321	192.168.1.6	80	Established
24.102.32.23	1025	192.168.1.6	80	Established
223.212.212	1046	192.168.1.6	80	Established

Table 2.3: Stateful Firewall Connection State Table

A stateful inspection firewall also differs from a packet filter firewall in that stateful inspection is useful or applicable only within TCP/IP network infrastructures. Stateful inspection firewalls can accommodate other network protocols in the same manner as packet filters, but the actual stateful inspection technology is relevant only to TCP/IP. For this reason, many texts classify stateful inspection firewalls as representing a superset of packet filter firewall functionality.

2.4. Application-Proxy Gateway Firewalls

Application-Proxy Gateway firewalls are advanced firewalls that combine lower layer access control with upper layer (Layer 7 – Application Layer) functionality.

Application-proxy gateway firewalls do not require a Layer 3 (Network Layer) route between the inside and outside interfaces of the firewall; the firewall software performs the routing. In the event the application-proxy gateway software ceases to function, the firewall system is unable to pass network packets through the firewall system. All network packets that traverse the firewall must do so under software (application-proxy) control.

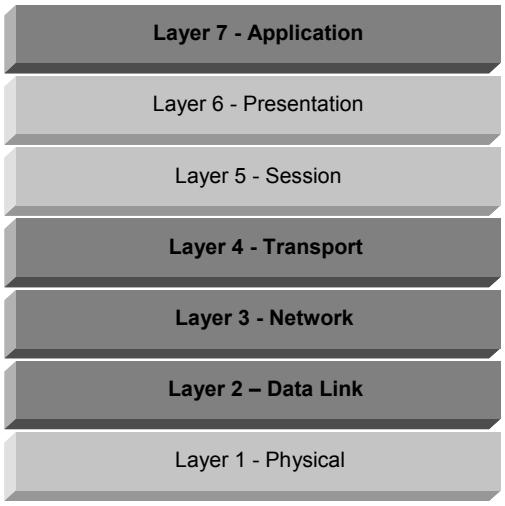

Figure 2.6: OSI Layers Addressed by Application-Proxy Gateway Firewalls

Each individual application-proxy, also referred to as a proxy agent, interfaces directly with the firewall access control ruleset to determine whether a given piece of network. traffic should be permitted 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 the following:

- User ID and Password Authentication,
- Hardware or Software Token Authentication,
- Source Address Authentication, and
- Biometric Authentication.

Application-proxy gateway firewalls have numerous advantages over packet filter firewalls and stateful inspection packet filter firewalls. First, application-proxy gateway firewalls usually have more extensive logging capabilities due to the firewall being able to examine the entire network packet rather than just the network addresses and ports. For example, application-proxy gateway logs can contain application-specific commands within the network traffic.

Another advantage is that application-proxy gateway firewalls allow security administrators to enforce whatever type of user authentication is deemed appropriate for a given enterprise infrastructure. Application-proxy gateways are capable of authenticating users directly, as opposed to packet filter firewalls and stateful inspection packet filter firewalls which normally authenticate users based on the network layer address of the system they reside on. Given that network layer addresses can be easily spoofed, the authentication capabilities inherent in application-proxy gateway architecture are superior to those found in packet filter or stateful inspection packet filter firewalls.

Finally, given that application-proxy gateway firewalls are not simply Layer 3 devices, they can be made less vulnerable to address spoofing attacks.

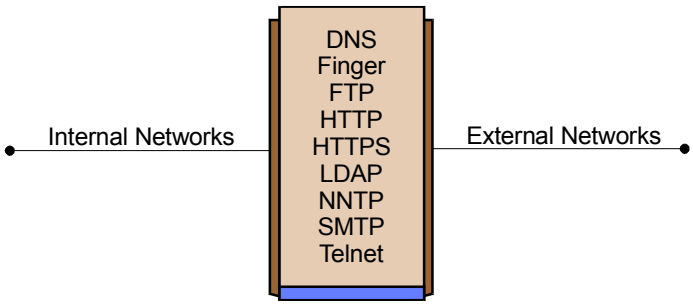

Figure 2.7: Typical Proxy Agents

The advanced functionality of application-proxy gateway firewalls also fosters several disadvantages when compared to packet filter or stateful inspection packet filter firewalls. First, because of the "full packet awareness" found in application-proxy gateways, the firewall is forced to spend quite a bit of time reading and interpreting each packet. For this reason, application-proxy gateway firewalls are not generally well suited to high-bandwidth or real-time applications. To reduce the load on the firewall, a dedicated proxy server (discussed in Section 2.5) can be used to secure less time-sensitive services such as email and most web traffic.

Another disadvantage is that application-proxy gateway firewalls 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. Most application-proxy gateway firewall vendors provide generic proxy agents to support undefined network protocols or applications. However, those generic agents tend to negate many of the strengths of the application-proxy gateway architecture and they simply allow traffic to "tunnel" through the firewall.

2.5. Dedicated Proxy Servers

Dedicated proxy servers differ from application-proxy gateway firewalls in that they retain proxy control of traffic but they do not contain firewall capability. They are typically deployed behind traditional firewall platforms for this reason. In typical use, a main firewall might accept inbound traffic, determine which application is being targeted, and then hand off the traffic to the appropriate proxy server, e.g., an email proxy server. The proxy server typically would perform filtering or logging operations on the traffic and then forward it to internal systems. A proxy server could also accept outbound traffic directly from internal systems, filter or log the traffic, and then pass it to the firewall for outbound delivery. An example of this would be an HTTP proxy deployed behind the firewall; users would need to connect to this proxy en route to connecting to external web servers. Typically, dedicated proxy servers are used to decrease the work load on the firewall and to perform more specialized filtering and logging that otherwise might be difficult to perform on the firewall itself.

As with application-proxy gateway firewalls, dedicated proxies allow an organization to enforce user authentication requirements as well as other filtering and logging on any traffic that traverses the proxy server. The implications are that an organization can restrict outbound traffic to certain locations or could examine all outbound email for viruses or

restrict internal users from writing to the organization's web server. Security experts have stated that most security problems occur from within an organization; proxy servers can assist in foiling internally based attacks or malicious behavior. At the same time, filtering outbound traffic will place a heavier load on the firewall and increase administration costs.

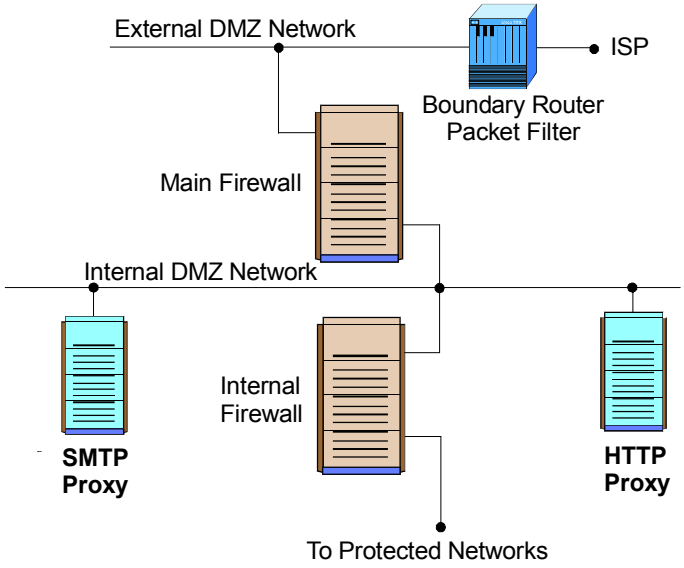

Figure 2.8: Application Proxy Configuration

In addition to authentication and logging functionality, dedicated proxy servers are useful for web and email content scanning, including the following:

- Java ™ applet or application filtering (signed versus unsigned or universal),
- ActiveX® control filtering (signed versus unsigned or universal),
- JavaScript filtering,
- Blocking specific Multipurpose Internet Multimedia Extensions (MIME) types – for example, "application/msword" for Microsoft® Word documents (see Section C.4 in Appendix C for suggestions for specific types),
- Virus scanning and removal,
- Macro virus scanning, filtering, and removal,
- Application-specific commands, for example, blocking the HTTP "delete" command, and
- User-specific controls, including blocking certain content types for certain users.

Figure 2.8 shows a sample diagram of a network employing dedicated proxy servers for HTTP and email placed behind another firewall system. In this case, the email proxy could be the organization's SMTP gateway for outbound email. The main firewall would hand off inbound email to the proxy for content scanning, and then the email could be made available to internal users by some means, e.g., POP or IMAP. 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, thereby reducing traffic on the firewall.

2.6. Hybrid Firewall Technologies

Recent advances in network infrastructure engineering and information security have caused a "blurring of the lines" that differentiate the various firewall platforms discussed earlier. As a result of these advances, firewall products currently incorporate functionality from several different classifications of firewall platforms. For example, many Application-Proxy Gateway firewall vendors have implemented basic packet filter functionality in order to provide better support for UDP (User Datagram) based applications.

Likewise, many packet filter or stateful inspection packet filter firewall vendors have implemented basic application-proxy functionality to offset some of the weaknesses associated with their firewall platform. In most cases, packet filter or stateful inspection packet filter firewall vendors implement application proxies to provide improved network traffic logging and user authentication in their firewalls.

Nearly all major firewall vendors have introduced hybridization into their products in some way, shape, or form, so it is not always a simple matter to decide which specific firewall product is the most suitable for a given application or enterprise infrastructure. Hybridization of firewall platforms makes the pre-purchase product evaluation phase of a firewall project important. Supported feature sets, rather than firewall product classification, should drive the product selection.

2.7. Network Address Translation

Network Address Translation (NAT) technology was developed in response to two major issues in network engineering and security. First, network address translation is an effective tool for "hiding" the network-addressing schema present behind a firewall environment. In essence, network address translation allows an organization to deploy an addressing schema of its choosing behind a firewall, while still maintaining the ability to connect to external resources through the firewall. Second, the depletion of the IP address space has caused some organizations to use NAT for mapping non-routable IP addresses to a smaller set of legal addresses, according to RFC 1918[9].

Network address translation is accomplished in three fashions:

Static Network Address Translation

In static network address translation, each internal system on the private network has a corresponding external, routable IP address associated with it. This particular technique is seldom used, due to the scarcity of available IP address resources. With static network

[9] RFC 1918 specifies several IP address ranges for Class A, B, and C networks. Addresses in these ranges can be used behind a firewall, but they cannot be routed on the Internet and therefore must be mapped to legal addresses.

address translation, it is possible to place resources behind (inside) the firewall, while maintaining the ability to provide selective access to external users. In other words, an external system could access an internal web server whose address has been mapped with static network address translation. The firewall would perform mappings in either direction, outbound or inbound. Table 2.4 shows an example of a static network address translation table that would map internal IP addresses, non-routable according to RFC 1918, to externally routable addresses.

Hiding Network Address Translation

With hiding network address translation, all systems behind a firewall share the same external, routable IP address. Thus, with a hiding network address translation system, five thousand systems behind a firewall will still look like only one system. This type of network address translation is fairly common, but it has one glaring weakness in that it is not possible to make resources available to external users once they are placed behind a firewall that employs it. Mapping in reverse from outside systems to internal systems is not possible, therefore systems that must be accessible to external systems must not have their addresses mapped. Another weakness of this particular network address translation implementation is that a firewall employing this type of network address translation must usually use its own external interface address as the "substitute" or translated address for all of the systems and resources that reside behind it. This requirement tends to impact the flexibility of this mechanism.

Internal (RFC 1918) IP Address	External (Globally Routable) IP Address
192.168.1.100	207.119.32.81
192.168.1.101	207.119.32.82
192.168.1.102	207.119.32.83
192.168.1.103	207.119.32.84
192.168.1.104	207.119.32.85
192.168.1.105	207.119.32.86
192.168.1.106	207.119.32.87
192.168.1.107	207.119.32.88
192.168.1.108	207.119.32.89
192.168.1.109	207.119.32.90

Table 2.4: Static Network Address Translation Table

Port Address Translation (PAT)

There are two main differences between PAT and Hiding NAT. First, PAT is not required to use the IP address of the external firewall interface for all network traffic; another address can be created for this purpose. Second, with port address translation, it is possible to place resources behind a firewall system and still make them selectively accessible to

external users. This access is accomplished by forwarding inbound connections on certain port numbers to specific hosts. For example, a firewall employing port address translation might pass all inbound connections to port 80 to an internal web server that employs a different (illegal, or RFC 1918) addressing schema.

Port address translation works by using the client port address to identify inbound connections. For example, if a system behind a firewall employing PAT were to telnet out to a system on the Internet, the external system would see a connection from the firewall's external interface, along with the client source port. When the external system replied to the network connection, it would use the above addressing information. When the PAT firewall received the response, it would look at the client source port provided by the remote system, and based on that source port, it would determine which internal system requested the session. In the example shown in Table 2.5, a remote system would respond to a connection request using the IP address of the external interface on the firewall, followed by the PAT Outbound Port as the client source port. The PAT Outbound Port is defined dynamically by the firewall itself, and it is sequential in some implementations and random (within the normal client source port parameters) in other implementations.

Internal System IP Address	Internal System Client Port	PAT Outbound Port
192.168.1.108	1028	3313
192.168.1.112	1039	3314
192.168.1.102	1400	3315
192.168.1.101	1515	3316
192.168.1.115	1027	3317
192.168.1.120	1026	3318

Table 2.5: Port Address Translation Table

In terms of strengths and weaknesses, each type of network address translation has applicability in certain situations, with the variable being the amount of design flexibility offered by each type. Static network address translation offers the most flexibility, but as stated earlier, static network address translation is not normally practical given the shortage of IP version 4 addresses. Hiding network address translation technology was an interim step in the development of network address translation technology, and is seldom used because port address translation offers additional features above and beyond those present in hiding network address translation while maintaining the same basic design and engineering considerations. PAT is often the most convenient and secure solution.

2.8. Host-Based Firewalls

Firewall packages are available in some operating systems such as Linux or as add-ons; they can be used to secure the individual host only. This can be helpful for use with internal servers; for example, an internal web server could be placed on a system running a host-based firewall. This carries several advantages, including the following:

- The server application is protected better than if it were running alone; internal servers should be protected and should not be assumed to be safe from attack because they are behind a main firewall.
- A separate firewall and subnet isn't necessary for securing the server; the host-based firewall performs these functions.

Host-based firewall packages typically provide access-control capability for restricting traffic to and from servers running on the host, and there is usually some limited logging available. While a host-based firewall is less desirable for high-traffic, high-security environments, in internal network environments or regional offices they offer greater security usually at a lower cost. A disadvantage to host-based firewalls is that they must be administered separately, and after a certain number it becomes easier and less expensive to simply place all servers behind a dedicated firewall configuration.

2.9. Personal Firewalls/Personal Firewall Appliances

Securing personal computers at home or remote locations is now as important as securing them at the office; many people telecommute or work at home and operate on organization- or agency-proprietary data. Home users dialing an Internet Service Provider (ISP), may have little firewall protections available to them because the ISP has to accommodate potentially many different security policies. Therefore, personal firewalls have been developed to provide protection for remote systems and to perform many of the same functions as larger firewalls.

These products are typically implemented in one of two configurations. One of these configurations is a *Personal Firewall*, which is installed on the system it is meant to protect; personal firewalls usually do not offer protection to other systems or resources. Likewise, personal firewalls do not typically provide controls over network traffic that is traversing a computer system – they only protect the computer system they are installed on.

The second configuration is called a *Personal Firewall Appliance*, which is in concept more similar to that of a traditional firewall. In most cases, personal firewall appliances are designed to protect small networks such as networks that might be found in home offices. These appliances usually run on specialized hardware and integrate some other form of network infrastructure components in addition to the firewall itself, including the following:

- Cable Modem WAN Routing,
- LAN Routing (dynamic routing support),
- Network hub,
- Network switch,
- DHCP (Dynamic Host Configuration Protocol) server,
- Network management (SNMP) agent, and
- Application-proxy agents.

Incorporating these infrastructure components into a firewall appliance allows an organization to deploy effective solutions consisting of a single piece of hardware.

Although personal firewalls and personal firewall appliances lack some of the advanced, enterprise-scale features of traditional firewall platforms, they can still form an effective piece of the overall security posture of an organization. In terms of deployment strategies, personal firewalls and personal firewall appliances normally address the connectivity concerns associated with telecommuters or branch offices. However, some organizations employ these devices on the organizational intranet, practicing a defense in depth strategy. Personal firewalls and personal firewall appliances can also be used to terminate VPNs: many vendors currently offering firewall-based VPN termination also offer a personal firewall client as well (see Section 3.3).

Management of the device or application is an important factor when evaluating or choosing a personal firewall/personal firewall appliance. Ideally, a personal firewall or personal firewall appliance should give the organization or agency the ability to enforce its defined security posture on all systems that connect to its networks and systems. In the case of telecommuters, this means that a personal firewall or personal firewall appliance should enforce a policy at least as restrictive as an end-user would experience if they were behind the corporate or agency firewall in the office.

Management of personal firewalls or personal firewall appliances should be centralized if possible. Again, centralization of management allows an organization or agency to enforce its security policy and posture on systems that are remotely connected. The best way to achieve this functionality is to create a security configuration profile that accompanies an end-user to any system logged into by that user. In this manner, the organization or agency's security policy will always be in effect when the user is accessing corporate or agency computing resources.

But what about remote users who connect to an organization's dial-in server and at other times connect to commercial ISPs? Assuming the security posture of the commercial ISP is less restrictive than the organization's, the risk of the computer being infected with a virus or other attack is greater, and connecting an infected computer to the organization's network could introduce the virus into that network. This is a problem, as many home users utilize their personal computers both for work and non-work related functions.

The ultimate solution is to use separate computers; for example, an organization could assign laptops to home users that can be used for work functions only and that cannot be connected to networks other than the organization's. This would include home networks as well. Each and every laptop should include a personal firewall and anti-virus software.

If such a solution isn't available, then the personal firewall must be in use at all times and must be configured to the most restrictive settings mandated by the organization. If, for example, Windows®-based file sharing is disabled by the firewall, it must remain disabled even when the computer is used for non-work functions. As well, if web security settings are set to reject certain types of content, this prohibition must remain in effect at all times. This policy has implications for the placement of the organization's dial-in server; it should be situated so that the firewall and proxies filter inbound traffic from dial-in connections. Essentially, a personal firewall, like anti-viral software, cannot protect a system if it is disabled or reconfigured at certain intervals with differing policies; it is an all or nothing proposition.

3. Firewall Environments

Firewall environment is a term used to describe the set of systems and components that are involved in providing or supporting the complete firewall functionality at a given point on a network. A simple firewall environment may consist of a packet filter firewall and nothing else. In a more complex and secure environment, it may consist of several firewalls, proxies, and specific topologies for supporting the systems and security. The following sections detail the systems and network topologies used in popular firewall environments.

3.1. Guidelines for Building Firewall Environments

There are four principles that should be noted before reading on, outlined in the following paragraphs:

Keep It Simple

The KISS principle is something that should be first and foremost in the mind of a firewall environment designer. Essentially, the more simple the firewall solution, the more secure it likely will be and the easier it will be to manage. Complexity in design and function often leads to errors in configuration.

Use Devices as They Were Intended to Be Used

Using network devices as they were primarily intended in this context means do not make firewalls out of equipment not meant for firewall use. For example, routers are meant for routing; their packet filtering capability is not their primary purpose, and the distinction should never be lost on those designing a firewall implementation. Depending on routers alone to provide firewall capability is dangerous; they can be misconfigured too easily. Network switches are another example (see Section 3.6); when used to switch firewall traffic *outside* of a firewall environment, they are susceptible to attacks that could impede switch functionality. In many cases, hybrid firewalls and firewall appliances are better choices simply because they are optimized to be firewalls first and foremost.

Create Defense in Depth

Defense in depth involves creating layers of security as opposed to one layer. The infamous Maginot line is, in hindsight, an excellent example of what not to do in firewall environments: place all your protection at the firewall. Where several firewalls can be used, they should be used. Where routers can be configured to provide some access control or filtering, they should be. If a server's operating system can provide some firewall capability, use it.

Pay Attention to Internal Threats

Lastly, attention to external threats to the exclusion of internal threats leaves the network wide open to attack from the inside. While it may be difficult to think of your work colleagues as posing a potential threat, consider that an intruder who gets past the firewall somehow could now have free reign to attack internal or external systems. Therefore, important systems such as internal web and email servers or financial systems should be placed behind internal firewalls or DMZ environments.

As a caveat to the above discussion, it should be noted that the expression, "all rules are meant to be broken," certainly applies when building firewall environments. Firewall designers should keep the above rules in mind when building environments, but every network and organization has its own unique requirements and idiosyncrasies, possibly requiring unique solutions.

3.2. DMZ Networks

The most common firewall environment implementation is known as a DMZ, or DeMilitarized Zone network. A DMZ network is created out of a network connecting two firewalls; i.e., when two or more firewalls exist in an environment, the networks connecting the firewalls can be DMZ networks.

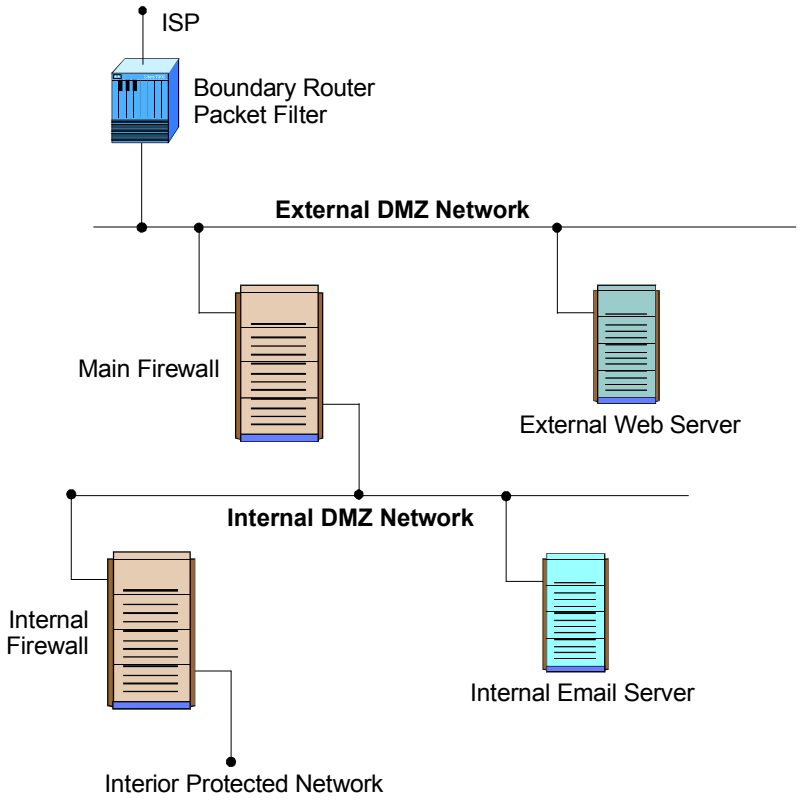

Figure 3.1: A DMZ Firewall Environment

DMZ networks serve as attachment points for computer systems and resources that need to be accessible either externally or internally, but that should not be placed on internal protected networks[10]. For example, an organization could employ a boundary router firewall and two internal firewalls, and place all externally accessible servers on the outer, or *external* DMZ between the router and the first firewall. The boundary router would filter packets

[10] See Section 3.4 in NIST Special Publication 800-10 for basic information on DMZ networks; DMZ networks are also referred to as *Screened Subnets*.

and provide protection for the servers, and the first firewall would provide access control and protection from the servers in case they were attacked. The organization could locate other internally accessible servers on the *internal* DMZ located between the two internal firewalls; the firewalls could provide protection and access control for the servers, protecting them both from external and internal attack. This environment is represented in Figure 3.1.

DMZ networks are typically implemented as network switches that sit between two firewalls or between a firewall and a boundary router. Given the special nature of DMZ networks, they typically serve as attachment points for systems that require or foster external connectivity. For example, it is often a good idea to place remote access servers and VPN endpoints in DMZ networks. Placing these systems in DMZ networks reduces the likelihood that remote attackers will be able to use them as vectors to enter private networks. In addition, placing these servers in DMZ networks allows the firewalls to serve as additional means for controlling the access rights of users that connect to these systems.

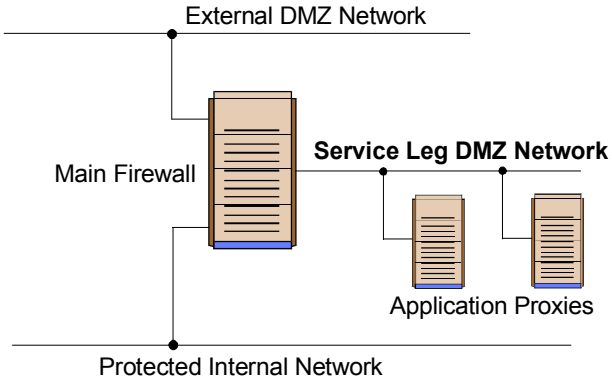

Figure 3.2: Service Leg DMZ Configuration

Service Leg Configuration

One DMZ network configuration is the so-called "service leg" firewall configuration, as shown in Figure 3.2. In the service leg configuration, a firewall is constructed with three different network interfaces. One network interface attaches to the boundary router, another network interface attaches to an internal connection point such as a network switch, and the third network interface forms the DMZ network. This configuration subjects the firewall to an increased risk of service degradation during a denial-of-service (DOS) attack aimed at servers located on the DMZ. In a standard DMZ network configuration, a denial-of-service attack against a DMZ-attached resource such as a web server will likely impact only that target resource. In a service-leg DMZ network configuration, the firewall bears the brunt of any denial-of-service attack because it must examine any network traffic before the traffic reaches the DMZ-attached resource. This can impact organizational traffic if, for example, the organization's popular web server is under attack.

3.3. Virtual Private Networks

Another valuable use for firewalls and firewall environments is the construction of Virtual Private Networks (VPNs). A virtual private network is constructed on top of existing net-

work media and protocols by using additional protocols and usually, encryption. If the VPN is encrypted, it can be used as an extension of the inner, protected network.

In most cases, virtual private networks are used to provide secure network links across networks that are not trusted. For example, virtual private network technology is increasingly used in the area of providing remote user access to organizational networks via the global Internet. This particular application is increasing in popularity due to the expenses associated with implementing private remote access facilities, such as modem pools. By using virtual private network technology, an organization purchases a single connection to the global Internet, and that connection is used to allow remote users access into otherwise private networks and resources. This single Internet connection can also be used to provide many other types of services. As a result, this mechanism is considered to be cost-effective.

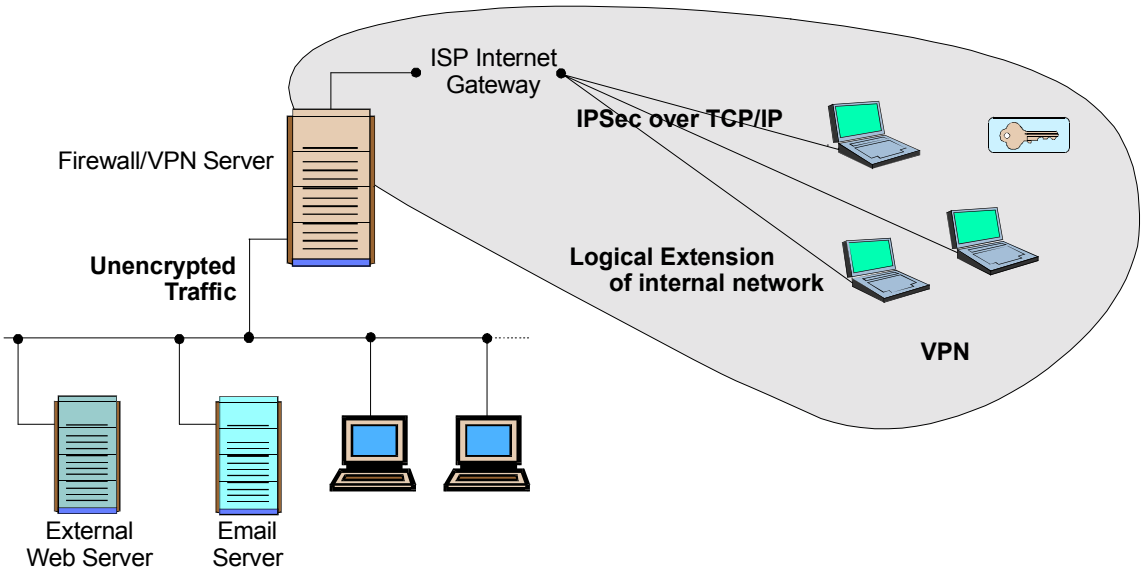

Figure 3.3: VPN Example

Virtual private network technology is often used to create secure networks between organizations or agencies, as shown in Figure 3.3.

On the protocol level, there are several possible choices for a modern virtual private network. The first, and perhaps the most currently used is a set of protocols known as IPSec[11] (Internet Protocol Security). The IPSec standards consist of IPv6 security features ported over to IPv4, the version of IP in use today on the Internet. Other current VPN protocols include PPTP (Point-to-Point Tunneling Protocol), a Microsoft standard, and the L2TP (Layer 2 Tunneling Protocol).

[11] See NIST ITL Bulletin *An Introduction to IPSec*, March 2001, at http://csrc.nist.gov

Placement of VPN Servers

In most cases, placing the VPN server at the firewall is the best location for this function. Placing it behind the firewall would require that VPN traffic be passed outbound through the firewall encrypted and the firewall is then unable to inspect the traffic, inbound or outbound, and perform access control, logging, or scanning for viruses, etc. Figure 3.3 shows a VPN that is terminated by the firewall, providing a logical extension of the internal protected network. The firewall employs IPSec between the remote laptop systems and presumably would pass the decrypted traffic between the firewall and the internal network.

Advanced virtual private network functionality comes with a price, however. For example, if VPN traffic is encrypted, there will be a decrease in performance commensurate with (a) the amount of traffic flowing across the virtual private network, and (b) the type/length of encryption being used. Performing encryption in hardware will significantly increase performance, however. For some DMZ environments, the added traffic associated with virtual private networks might require additional capacity planning and resources.

3.4. Intranets

An intranet is a network that employs the same types of services, applications, and protocols present in an Internet implementation, without involving external connectivity. For example, an enterprise network employing the TCP/IP protocol suite, along with HTTP for information dissemination would be considered an Intranet. In Figure 3.4, the internal protected networks are examples of intranet configurations.

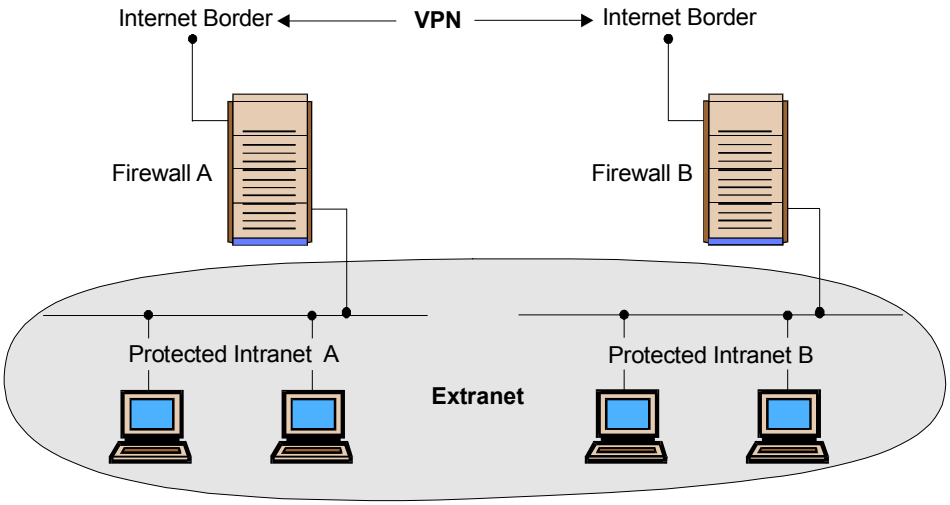

Figure 3.4: VPN/Extranet Joining Two Intranets

Most organizations currently employ some type of intranet, although they may not refer to the network as such. Within the internal network (intranet), many smaller intranets can be created by the use of internal firewalls. As an example, an organization may protect its personnel network with an internal firewall, and the resultant protected network may be referred to as the personnel intranet.

Since intranets utilize the same protocols and application services present on the Internet, many of the security issues inherent in Internet implementations are also present in intranet implementations. Therefore, intranets are typically implemented behind firewall environments.

3.5. Extranets

Extranets form the third piece of the modern enterprise connectivity picture. An extranet is usually a business-to-business intranet; that is, two intranets are joined via the Internet. The extranet allows limited, controlled access to remote users via some form of authentication and encryption such as provided by a VPN.

Extranets share nearly all of the characteristics of intranets, except that extranets are designed to exist outside a firewall environment. By definition, the purpose of an extranet is to provide access to potentially sensitive information to specific remote users or organizations, but at the same time denying access to general external users and systems. Extranets employ TCP/IP protocols, along with the same standard applications and services.

Many organizations and agencies currently employ extranets to communicate with clients and customers. Within an extranet, options are available to enforce varying degrees of authentication, logging, and encryption. Figure 3.4 shows an example topology of an extranet.

3.6. Infrastructure Components: Hubs and Switches

In addition to routers and firewalls, infrastructure devices such as hubs and switches provide connectivity between systems. The most simple of these connection devices is the network concentrator, or hub. Hubs are devices that function at Layer 1 of the OSI model. In other words, there is no real intelligence in network hubs; they exist only to provide physical attachment points for networked systems or resources.

There are numerous weaknesses associated with network hubs. First and foremost, network hubs allow any device connected to them to see the network traffic destined for, or originating from, any other device connected to that same network hub. For this reason, network hubs should not be used to build DMZ networks or firewall environments.

A more advanced infrastructure device is the network switch. Network switches are Layer 2 devices, which means that they actually employ basic intelligence in providing attachment points for networked systems or components. Network switches are essentially multiport bridges, so they are also capable of delivering the full network bandwidth to each physical port. Another side effect of the bridging nature of switches is that systems connected to a switch cannot eavesdrop on each other. These anti-eavesdrop capabilities inherent in network switches make them useful for implementing DMZ networks and firewall environments.

It is important to note that switches should not be used to provide any firewall or traffic isolation capability outside of a firewall environment, due to denial of service-like attacks that can cause switches to flood connected networks with packets. Also, the inherent capability of network switches, that is, providing subnet isolation, can also affect how Intrusion Detection Systems (IDS) must be deployed and implemented.

3.7. Intrusion Detection Systems

Intrusion Detection Systems (IDS)[12] are designed to notify and in some cases prevent unauthorized access to a networked system or resource. Many intrusion detection systems are also capable of interacting with firewalls in order to bring a reactive element to the provision of network security services. Firewalls that interact with intrusion detection systems are capable of responding to perceived remote threats automatically, without the delays associated with a human response. For example, if an intrusion detection system detects a denial-of-service attack in progress, it can instruct certain firewalls to automatically block the source of the attack (albeit, false positives responses can occur).

Host-Based IDS

Two different types of intrusion detection systems are generally available. The first type, host-based intrusion detection, must be installed on each individual computer system that is to be protected. Host-based intrusion detection is very closely integrated with the operating system it protects, so each different operating system will have a different host-based intrusion detection module. Host-based intrusion detection systems, therefore, are usually able to detect threats at a high level of granularity. Weaknesses associated with host-based intrusion detection include:

- Often, host-based intrusion detection products have a negative impact on system performance. The larger the number of parameters examined by the intrusion detection system, the greater the impact on system performance.
- Host-based intrusion detection systems do not always notice network-based attacks such as denial of service.
- Many host-based intrusion detection systems have a negative impact on operating system stability.

Network-Based IDS

The second type of intrusion detection system is network-based intrusion detection. Network-based intrusion detection systems are implemented as protocol analyzers with intelligence. These devices monitor network traffic that "passes by" on the wire, looking for "attack signatures" that indicate certain types of attacks are in progress. Attack signatures are simply strings of characters that are often present during an attack. Network-based intrusion detection is normally more effective than host-based intrusion detection due to the fact that a single system can monitor multiple systems and resources (albeit host-based is more appropriate for monitoring a specific system). Issues associated with network-based intrusion detection include:

- Many network-based intrusion systems miss attack signatures that are spread across multiple packets. Most network-based intrusion detection systems do not have the capability of reassembling all fragmented network traffic. This can be used to bypass network-based intrusion detection systems.

---

[12] See NIST Special Publication 800-31, *Intrusion Detection Systems*, at http://csrc.nist.gov

- Network-based intrusion detection systems rely on promiscuous mode network interfaces to examine all network traffic on a given wire. If proper network security guidelines are followed (i.e., use switches instead of hubs for network attachment points), network-based intrusion detection systems cannot function without special switch configurations (port mirroring, etc.). Many network switches lack such functionality.
- Most network-based intrusion detection systems can be detected using tools designed to locate/identify promiscuous mode interfaces. Once the promiscuous mode interface has been detected, it is not normally difficult to crash the intrusion detection system or to flood it with useless network traffic.
- Many intrusion detection systems lack the functionality necessary to identify network-layer attacks. Basically, not all attacks will have a predictable attack signature.
- In the context of denial-of-service attacks, many intrusion detection systems are disabled by the very events they are supposed to monitor.

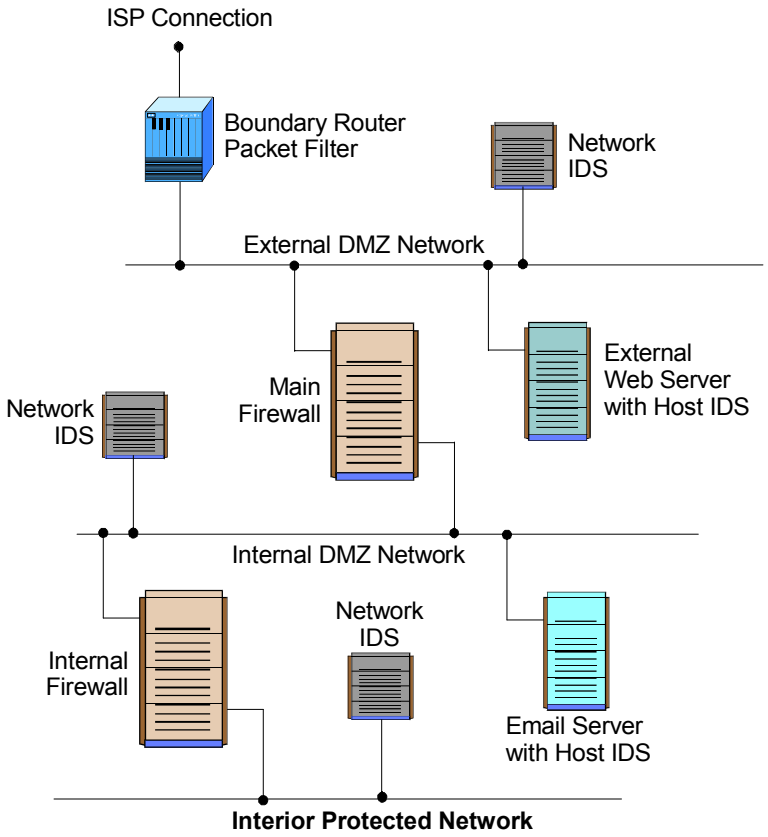

Figure 3.5: IDS Placement Throughout a Network

Users should be aware that most existing types of intrusion detection are not difficult to bypass if the attacker is knowledgeable. In addition, users should be aware that intrusion detection systems generate voluminous logs that must be examined carefully if the intrusion detection system is to be effective. Also, the handling of false-positive notifications is important; automated systems are prone to mistakes, and human differentiation of possible attacks is resource-intensive. It is therefore important to consider continuous fine-tuning of IDS implementations to make them manageable when enforcing compliance with an organ-

izational security policy while at the same time providing meaningful data on which to base decisions.

Organizations must have a thorough understanding of the flow of data across their networks and systems to properly implement an intrusion detection system solution. It is advisable to place host-based intrusion detection tools on all mission-critical systems, even those that should not, in theory, allow external access. By placing agents on these systems, organizations are better able to notice a security incident in progress. It is important to place intrusion detection systems at any location where network traffic from external entities is allowed to enter controlled or private networks. For example, many organizations that have Internet connectivity choose to implement network-based intrusion detection systems in their DMZ networks as well as behind firewalls, as shown in Figure 3.5.

3.8. Domain Name Service (DNS)

The Domain Name Service (DNS) is critical to any environment that makes use of the Internet. Because of the sensitive nature of this service, special security measures are warranted.

First, internal domain name servers should be kept separate from external domain name servers. For example, a domain name server that is accessible to the entire world should not contain entries for systems that cannot be reached from the outside world, with the possible exception being authenticated remote users. Allowing such private entries to exist in an external domain name server only serves to provide a target list for a remote attacker. An organization should maintain separate internal and external domain name servers. This practice, known as *split DNS*, ensures that private internal systems are never identified to persons external to the organization.

Second, it is also necessary to control the types of access any given domain name server will allow. Basically, the domain name service application can operate using two different IP transports: user lookups employ the User Datagram Protocol (UDP), and domain name server-to-server communication employs the transmission control protocol (TCP). Domain name service connections using the transmission control protocol are also known as zone transfers. Access to a domain name server using the transmission control protocol should be restricted to only those domain name servers that are under the direct control of the organization. The primary risk with allowing blind zone transfers is that of modifying domain name service information. For example, if a server allows blind or unrestricted zone transfers, it is possible for a remote attacker to modify the domain name service information on that server in order to redirect network traffic away from a legitimate site.Figure 3.6 shows a split DNS example. The internal DNS server would be set up to resolve (find) names for internal systems, so that internal systems could connect to other internal systems, all systems on the DMZ, and the rest of the Internet. The external DNS server would permit external systems to resolve names for the main firewall, itself, and systems on the external DMZ, but not the internal network. As a result, these systems only would be visible to the rest of the Internet.

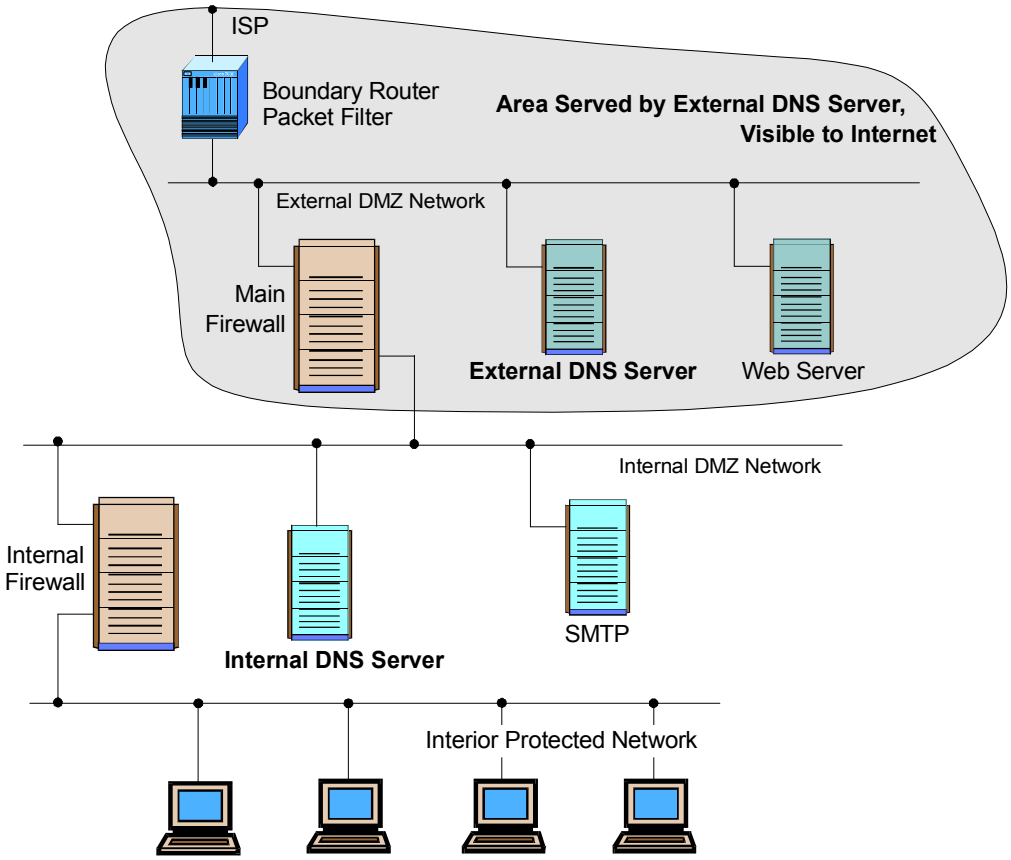

Figure 3.6: Split DNS example

3.9. Placement of Servers in Firewall Environments

Where to place servers in a firewall environment depends on many factors, including the number of DMZs, the external and internal access required for the servers located on the DMZ, the amount of traffic, and the sensitivity of the data served. It is not possible to proscribe a "one size fits all" recommendation for server location, but several guidelines can be used to make the determination, including the following:

- Protect external servers with a Boundary Router/Packet Filter.
- Do not place externally accessible servers on the protected network.
- Place internal servers behind internal firewalls as their sensitivity and access require.
- Isolate servers such that attacks on the servers do not impair the rest of the network.

The following paragraphs contain some suggestions for locating specific servers and systems. While the location of servers will be determined by each organization's specific requirements, every effort should be made to provide protection for the servers both from out-

side and inside threats, and to isolate attacks on the servers so that the rest of the organization is not affected.

Externally Accessible Servers

Externally accessible web servers, as well as directory servers or DNS servers, can be placed on an external DMZ, that is, between a boundary router and a main firewall. The boundary router can provide some access control and filtering for the servers, and the main firewall can restrict connections from the servers to internal systems, which could occur if the servers are penetrated. In the case of popular, heavily used servers, a high-speed boundary router with several DMZ attachments could be used to isolate the server(s) on individual DMZ networks. Thus, if a DDOS attack is mounted against a server, the rest of the network would not suffer.

VPN and Dial-in Servers

These servers are better placed on an external DMZ so that their traffic passes through the firewall. One suggested configuration is to place the VPN server on the firewall platform, so that outbound traffic can be encrypted *after* it has been filtered (e.g., by an HTTP proxy) and inbound traffic can be decrypted and again, filtered by the firewall. The dial-in server should be placed on an external DMZ for the same reasons.

Internal Servers

Internally accessible web servers, email servers, and directory servers can be placed on an internal DMZ, that is, between two dedicated firewalls, the main and the internal, with the internal firewall separating the DMZ from the protected network. Placing these systems on an internal DMZ provides defense in depth protection from external threats, and provides protection from internal threats. If an HTTP proxy is used for outbound HTTP traffic, placing this system on the internal DMZ provides more protection from insider/external threats.

Mail Servers

Some firewalls can be used to accept email, that is, SMTP connections. A popular configuration includes using the main firewall to (a) accept SMTP connections and (b) then pass them off to a dedicated proxy/email server located on the internal DMZ. This eliminates the need for the firewall to process the email for active content and attachments.

If users need to access email from external networks, for example when on travel or at conferences, one method for protecting the organizational email server from direct external access is to run an SSL proxy on the main firewall. Using a web browser, external users would connect to the main firewall (the main firewall could be configured with an alias to disguise its name). The main firewall would forward the SSL connection to the internal proxy/email server, which would serve the email over the web. The solution prevents direct external access to the mail server, yet still permits external access through the firewall. This approach could be used for other types of servers as well.

As a summary, Figure 3.7, below, shows an example firewall environment with an external and internal DMZ and several servers and intrusion detection devices. In this example, the VPN server is combined with the main firewall and the dial-in server is located between the boundary router/packet filter and the main firewall. Other externally accessible servers are located on the external DMZ as well. All other internal servers are located on the internal DMZ, protected both from external *and* internal threats.

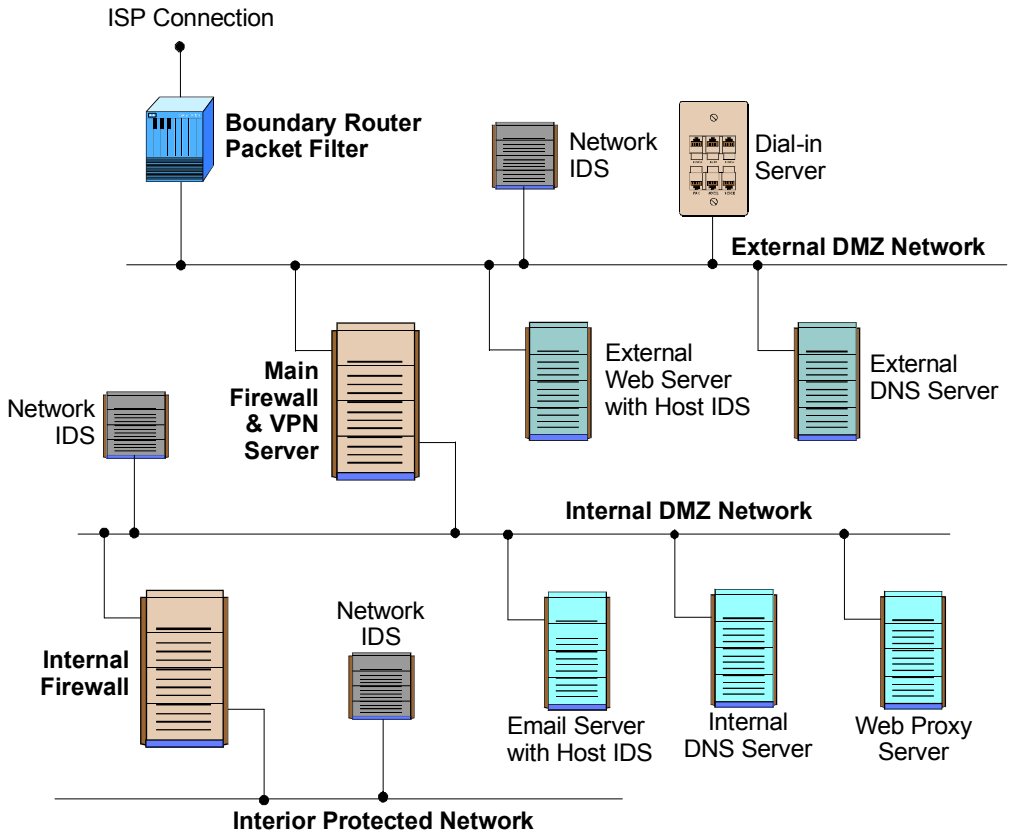

Figure 3.7: Summary Example Firewall Environment

4. Firewall Security Policy

A specific and strongly worded information security policy is vital to the pursuit of external connectivity and commerce. This policy should govern everything from acceptable use to response scenarios in the event a security incident occurs. A firewall policy is distinct from the information security policy, in as much as it is simply a description of how the information security policy will be implemented by the firewall and associated security mechanisms.

Without a firewall policy, administrators and organizations are "flying blind." Firewalls can be complex and tricky to manage, and security incidents can occur daily. Without a policy to guide firewall implementation and administration, the firewall itself may become a security problem. This section presents steps for creating a firewall policy and then follows up with an example. It contains recommendations for testing the policy and periodically updating the policy.

4.1. Firewall Policy

A firewall policy dictates how the firewall should handle applications traffic such as web, email, or telnet. The policy should describe how the firewall is to be managed and updated.

Before a firewall policy can be created, some form of risk analysis must be performed on the applications that are necessary for accomplishment of the organization's mission. The results of this analysis will include a list of the applications and how those applications will be secured. The process to create this list is not detailed here[13], however, it will require knowledge of the vulnerabilities associated with each application and the cost-benefits associated with the methods used for securing the applications. Risk analysis of the organization's information technology infrastructure should be weighed based on an evaluation of the following elements: threats, vulnerabilities, and countermeasures in place to mitigate vulnerabilities, and the impact if sensitive data is compromised. The goal is to understand and evaluate these elements prior to establishing a firewall policy.

The result of the risk analysis will dictate the manner in which the firewall system handles network applications traffic. The details of which applications can traverse a firewall, and under what exact circumstances such activities can take place, should be documented in the form of an applications traffic matrix, as shown in Table 4.1.

The steps involved in creating a firewall policy are as follows:

- Identification of network applications deemed necessary,
- Identification of vulnerabilities associated with applications,
- Cost-benefits analysis of methods for securing the applications,
- Creation of applications traffic matrix showing protection method, and

[13] See NIST Special Publications 800-30, *Risk Management*, and 800-18, *Guide for Developing Security Plans for Information Technology Systems*, at http://csrc.nist.gov

- Creation of firewall ruleset based on applications traffic matrix.

TCP/IP APPLICA-TIONSERVICE	LOCATION	INTERNAL HOST TYPE	INTERNAL HOST SECURITY POLICY	FIREWALL SECURITY POLICY (Internal)	FIREWALL SECURITY POLICY (External)
Finger	Any	Unix	TCP Wrapper	Permit	Reject
"	Any	PC - TCP/IP	None	Permit	Permit
FTP	Any	Unix	No Anonymous; UserID/Password; Secure Shell (SSH)	Permit	Application Proxy with User Authenti-cation
"	Any	PC - TCP/IP	Client Only; Anti-Virus	Permit	Application Proxy with User Authenti-cation
TFTP	Any	Unix Server with Diskless Clients Only	Secure Mode; Permit tftp to Limited Direc-tories	Permit Only Local Domain; Reject Other	Reject
"	Any	Unix - All Other	Disable	Reject	Reject
"	Any	PC - TCP/IP	Disable	Reject	Reject
Telnet	Any	Unix	Secure Shell	Permit	Application Proxy with User Authenti-cation
"	Any	PC - TCP/IP	Client Only	Permit	Application Proxy with User Authenti-cation
"	Any	Router/Firewall	2 Password Layers; Token Authentication	Token Authentica-tion	Reject
NFS	Any	UNIX	Limit Exports; Host/Groups (Granu-lar Access)	Reject All, except by Written Authori-zation	Reject
"	Any	PC - TCP/IP	Client Only	Reject	Reject
NetBIOS over TCP/IP	Any	Windows NT/95/WFW	Limit Access to Shares	Permit Local Do-main Only; Reject Others	Reject

Table 4.1: Firewall Application Traffic Ruleset Matrix

4.2. Implementing a Firewall Ruleset

Most firewall platforms utilize rulesets as their mechanism for implementing security controls. The contents of these rulesets determine the actual functionality of a firewall. De-

pending on the firewall platform architecture, firewall rulesets can contain various pieces of information. Nearly all rulesets, however, will contain the following fields, as a minimum:

- The *source address of the packet*, i.e., the Layer 3 address of the computer system or device the network packet originated from (an IP address such as 192.168.1.1).
- The *destination address of the packet*, in other words, the Layer 3 address of the computer system or device the network packet is trying to reach (e.g., 192.168.1.2).
- The *type of traffic*, in other words, the specific network protocol being used to communicate between the source and destination systems or devices – often Ethernet at Layer 2 and IP at Layer 3.
- Possibly some *characteristics of the Layer 4 communications sessions* – the protocol such as TCP, and the source and destination ports of the sessions (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).
- Sometimes, *information pertaining to which interface of the router the packet came from* and which interface of the router the packet is destined for – useful for routers with three or more network interfaces.
- An action, such as *Deny* or *Permit* the packet, or *Drop* the packet, which does not return a response to the packet's sender as does Deny.

Users should be aware that firewall rulesets tend to 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). That same firewall ruleset will likely contain many more rules by the time the firewall system reaches the end of its first year in production. New user or business requirements typically drive these changes, but they can also reflect political forces within an organization or agency.

The firewall ruleset can be assembled after completing the applications traffic matrix. Depending on the firewall, this may be done through a web-style interface; in the case of a packet filter, it may be done manually. Firewall rulesets should be built to be as specific as possible with regards to the network traffic they control. Rulesets should be kept as simple as possible, so as not to accidentally introduce "holes" in the firewall that might allow unauthorized or unwanted traffic to traverse a firewall.

The default policy for the firewall for handling inbound traffic should be to block all packets and connections unless the traffic type and connections have been specifically permitted. This approach is more secure than another approach used often: permit all connections and traffic by default and then block specific traffic and connections.

The firewall ruleset should always block the following types of traffic:

- *Inbound traffic from a non-authenticated source system with a destination address of the firewall system itself*. This type of packet normally represents some type of probe or attack against the firewall. One common exception to this rule would be in the event the firewall system accepts delivery of inbound email (SMTP on port 25). In this event, the firewall must allow inbound connections to itself, but only on port 25.

- Inbound traffic with a source address indicating that the packet originated on a network behind the firewall. This type of packet likely represents some type of spoofing attempt.
- *Inbound traffic containing ICMP (Internet Control Message Protocol) traffic.* Since ICMP can be used to map the networks behind certain types of firewalls, ICMP should not be passed in from the Internet, or from any untrusted external network.
- Inbound or Outbound traffic from a system using a source address that falls within the address ranges set aside in RFC 1918 as being reserved for private networks. For reference purposes, RFC 1918 reserves the following address ranges for private networks:

  10.0.0.0 to 10.255.255.255 (Class A, or "/8" in CIDR[14] notation)

  172.16.0.0 to 172.31.255.255 (Class B, or "/12" in CIDR notation)

  192.168.0.0 to 192.168.255.255 (Class C, or "/16" in CIDR notation)

  Inbound traffic with these source addresses typically indicates the beginning of a denial-of-service attack involving the TCP SYN flag. Some firewalls include internal functionality to combat these attacks, but this particular type of network traffic should still be blocked with ruleset entries.

- *Inbound traffic from a non-authenticated source system containing SNMP (Simple Network Management Protocol) traffic.* These packets can be an indicator that an intruder is probing a network, but there are few reasons an organization or agency might want to allow inbound SNMP traffic, and it should be blocked in the vast majority of circumstances.
- *Inbound traffic containing IP Source Routing information.* Source Routing is a mechanism that allows a system to specify the routes a piece of network traffic will employ while traveling from the source system to the destination system. From a security standpoint, source routing has the potential to permit an attacker to construct a network packet that bypasses firewall controls. In modern networks, IP Source Routing is rarely used, and valid applications are even less common on the Internet.
- Inbound or Outbound network traffic containing a source or destination address of 127.0.0.1 (localhost). Such traffic is usually some type of attack against the firewall system itself.
- *Inbound or Outbound network traffic containing a source or destination address of 0.0.0.0.* Some operating systems interpret this address as either localhost or as a broadcast address, and these packets can be used for attack purposes.
- *Inbound or Outbound traffic containing directed broadcast addresses.* A directed broadcast is often used to initiate a broadcast propagation attack such as SMURF[15]. Di-

---

[14] CIDR is short for Classless Inter-Domain Routing, an IP addressing scheme that replaces the scheme based on classes A, B, and C. CIDR addresses reduce the size of routing tables and make more IP addresses available within organizations. CIDR was created to help reduce problems associated with IP address depletion.

[15] See NIST ITL Bulletins Computer Attacks: What They Are and How to Defend Against Them, May 1999, and Mitigating Emerging Hacker Threats, June, 2000, at http://csrc.nist.gov

rected broadcasts allow one computer system to send out a broadcast message with a source address other than its own. In other words, a system sends out a broadcast message with a spoofed source address. 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 that has been used to disable some of the largest sites on the Internet.

Some types of firewalls are also capable of integrating user authentication into ruleset enforcement. For example, many firewalls have the capability of blocking access to certain systems until a user authenticates to the firewall. This authentication can be internal to the firewall or external to the firewall. Firewalls that implement application proxies can also integrate with advanced enterprise authentication schemes.

Most firewalls also support multiple options for logging. These options range anywhere from the creation of simple log entries, up to options for alerting users that a certain event has occurred. Depending on the alert implementation, this action can include a range of options, from sending email notification, to paging appropriate personnel.

4.3. Testing Firewall Policy

Policies are implemented every day but these policies are rarely checked and verified. For nearly all companies or agencies, firewall and security policies should be audited and verified at least quarterly.

In many cases, firewall policy can be verified using one of two methodologies. The first methodology, and by far the easiest, is to obtain hardcopies of the firewall configurations and compare these hardcopies against the expected configuration based on defined policy. All organizations, at a minimum, should utilize this type of review.

The second methodology involves actual in-place configuration testing. In this methodology, the organization utilizes tools that assess the configuration of a device by attempting to perform operations that should be prohibited. Although these reviews can be completed with public-domain tools, many organizations, especially those subject to regulatory requirements, will choose to employ commercial tools.

While the second methodology is more rigorous, both methodologies should be employed. The goal is to make sure that the firewalls (as well as any other security-related devices) are configured exactly as they should be, based upon the written policy. It is also important that the firewall system itself be tested using security assessment tools. These tools should be used to examine the underlying firewall operating system, as well as the firewall software and implementation. As before, these assessment tools can be public domain or commercial (or both).

4.4. Firewall Implementation Approach

When implementing firewalls and firewall policy, organizations must decide whether to implement the firewall as an appliance or on top of a commercial operating system. While this decision will be largely determined by organization or agency requirements, the following issues should be considered:

First, in general terms, appliance-based firewalls will be more secure than those implemented on top of commercial operating systems. Appliance-based firewalls do not suffer from security vulnerabilities associated with underlying operating systems. Appliance-based firewalls generally employ ASIC (Application-Specific Integrated Circuit) technology, with the actual firewall software being present as firmware driving the ASICs. These firewalls also tend to be faster than firewalls implemented on top of commercial operating systems.

The advantage of implementing firewalls on top of commercial operating systems is scalability. If an environment requires improved performance, organizations can buy a larger system on which to run the firewall software. Most appliances do not offer this level of flexibility or scalability.

The greatest disadvantage of implementing firewalls on top of commercial operating systems is the potential presence of vulnerabilities that might undermine the security posture of the firewall platform itself. In most circumstances where commercial firewalls are breached, that breach is facilitated by vulnerabilities in the underlying operating system[16]. Much expertise is needed in securing the underlying operating system and maintaining it.

This decision must be made based on relative costs, as well as estimates of future requirements.

4.5. Firewall Maintenance & Management

Commercial firewall platforms employ one of two mechanisms for configuration and ongoing maintenance. The first mechanism is command-line interface (CLI) configuration, which enables an administrator to configure the firewall by typing commands into a command prompt. This technique is error-prone due to typing mistakes, however. The primary advantage to command-line configuration is that a skilled and experienced administrator can configure the firewall and react to emergency situations more quickly than with a graphic interface.

The second (and most common) mechanism for firewall configuration is through a graphic user interface. Graphic interfaces are simpler and enable a novice administrator to configure advanced systems in a reasonable amount of time. The major issue with graphic interfaces is configuration granularity. In many modern firewall platforms, there are options available in the firewall that cannot be configured using the graphic interface. In these circumstances, a command-line interface must be used.

For either option, great care must be taken to ensure that all network traffic dealing with firewall system management is secured. For web-based interfaces, this security will likely be implemented through Secure Sockets Layer[17] (SSL) encryption, along with a user ID and

---

[16] NIST has produced a database of vulnerabilities associated with a wide variety of different operating systems and security products. This database can be searched easily to find problems and their associated patches. See http://icat.nist.gov

[17] The Secure Sockets Layer (SSL) is based on public key cryptography; it is used to generate a cryptographic session key that is private to a web server and a client browser and that cannot be duplicated by a third party. The communications session is encrypted and therefore private; many uses of SSL

password. For proprietary (non-web) interfaces, custom transport encryption is usually implemented. It should be a matter of policy that all firewall management functions take place over secured links using strong authentication and encryption.

4.6. Physical Security Of The Firewall Environment

The physical security of the firewall, for the firewall environment, is sometimes overlooked. If the devices are located in a nonsecure area, they are susceptible to damage from intruders and at a higher risk to accidental damage. Therefore, firewall devices should be secured behind locked doors. Some organizations locate their firewall environments in secured computing facilities, complete with guards and other physical security alarms.

Another factor in physical security is the quality of the electrical and network connections and environment control. The firewall facility should have backup power supplies and possibly redundant connections to external networks. Some form of air-conditioning and air filtration is also typically a requirement.

Lastly, the firewall facility should be protected, as is reasonable, from natural disasters such as fire and flood. Fire suppressant systems are usually standard equipment in computing facilities.

4.7. Periodic Review Of Information Security Policies

As with any type of policy, information security policies must undergo periodic review in order to ensure accuracy and timeliness. Best practice dictates that information security policies should be reviewed and updated at least twice per year. Best practice further dictates that several events can trigger a review of information security policies. These triggers include events such as the implementation of major enterprise computing environment modifications and any occurrence of a major information security incident.

A formal approach for managing which services are allowed through the firewall should be implemented. For example, when new applications are being considered, a configuration control board could evaluate new services before the firewall administrators are formally notified to implement the service. Alternatively, when an application is phased out or upgraded, the firewall ruleset should be formally changed. This approach adds some rigor and discipline to the firewall policy implementation, minimizing the presence of old and potentially insecure rules that are no longer needed.

Firewall installations as well as systems and other resources must be audited on a regular, periodic basis. In some cases, these periodic reviews can be conducted on paper by reviewing hardcopy configurations provided by appropriate systems administration staff. In other cases, periodic reviews should involve actual audits and vulnerability assessments of production and backup infrastructure components, computer systems, and other various types of resources.

---

are for secure financial transactions in which credit card information must be kept private from potential third-party observers of communications traffic.

It is equally important that companies or agencies with Internet connectivity employ additional measures to ensure the overall security of these environments. These specialized audits or assessments are known as penetration analyses. Penetration analyses should be employed in addition to, not instead of, a conventional audit program. Penetration analyses can be either "seeded" or "blind," depending on the circumstances involved.

A seeded penetration is a penetration analysis in which the organization or team conducting the assessment has been provided with detailed network and system information prior to the execution of the assessment. Because this type of assessment does not require any advanced discovery techniques on the part of the entities executing the test, this type of test is typically conducted by entities that lack the expertise to conduct a blind penetration. Also, a seeded penetration might be employed when an organization or agency wants to limit the scope of an analysis to a given environment or set of systems.

A blind penetration is an assessment where minimal information exchange occurs prior to the beginning of the assessment. It is therefore up to the organization or team conducting the assessment to obtain all information relevant to the conduct of the assessment, within the time constraints of the assessment. This initial discovery effort makes a blind penetration analysis much more difficult than a seeded penetration. Likewise, the results of a blind penetration are much more realistic and dramatically more indicative of the actual level of risk associated with global connectivity.

4.8. A Sample Topology and Ruleset

This section presents a sample firewall topology and ruleset based the following requirements:

- All internal network traffic permitted outbound to all sites through both firewalls and the boundary router,
- Inbound SMTP (email) permitted to the main firewall where it is passed to a proxy server and then to internal email clients,
- Outbound HTTP (web) traffic permitted to the internal firewall where it is passed to an HTTP proxy server, and then onto external websites,
- Inbound connections from remote systems permitted to the firewall's VPN port where it is passed to internal systems, and
- All other inbound traffic blocked.

In reality this list would be longer and more specific. In this example, the HTTP application proxy could cache web pages for performance reasons, and it could also filter active content such as Java™, JavaScript, or ActiveX® controls and log outbound connections. The SMTP application proxy would examine all email attachments or in-line content for viruses and quarantine the infected code as necessary.

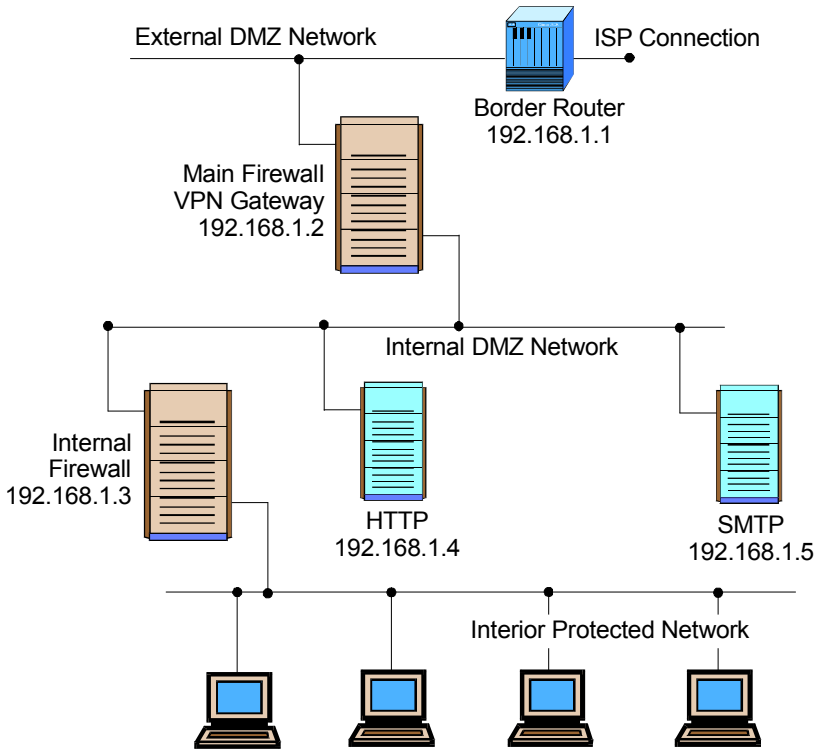

Figure 4.1: Sample Firewall Environment

The firewall environment for this network is shown in Figure 4.1. An external DMZ network would connect to the Internet via a packet filter serving as a boundary router – Section 2.2 detailed reasons why using a packet filter is preferable. The main firewall would incorporate a VPN port for remote users; such users would need VPN client software to connect to the firewall. Email inbound would connect to the main firewall first, which would pass it on to an application proxy server located on an internal DMZ. Outbound web traffic would connect to the internal firewall, which would pass it on to an HTTP application proxy located on the internal DMZ.

A ruleset for the boundary router would look as follows, in Table 4.2. It contains the default blocking rules described as in Section 4.2. Note: This ruleset is greatly simplified; a real example would involve vendor-specific conventions and other details.

Rule 1 allows return packets from established connections to return to the source systems (note that if the boundary router was a hybrid stateful firewall, rule 1 would not be necessary). Rule 3 permits inbound connections to the main firewall's VPN port; rules 4 and 5 tell the router to pass SMTP and HTTP traffic to the main firewall, which will send the traffic to the respective application proxies. Rule 8 then denies all other inbound connections to the main firewall (or any other systems possibly located on the external DMZ).

	Source Address	Source Port	Destination Address	Destination Port	Action	Description
1	Any	Any	192.168.1.0	> 1023	Allow	Rule to allow return TCP Connections to internal subnet
2	192.168.1.1	Any	Any	Any	Deny	Prevent Firewall system itself from directly connecting to anything
3	Any	Any	192.168.1.2	VPN	Allow	Allow External users to connect to VPN server
4	Any	Any	192.168.1.2	SMTP	Allow	Allow External Users to send email to proxy
5	Any	Any	192.168.1.2	HTTP	Allow	Send inbound HTTP to proxy
6	Any	Any	192.168.1.1	Any	Deny	Prevent External users from directly accessing the Firewall system.
7	192.168.1.0	Any	Any	Any	Allow	Internal Users can access External servers
8	Any	Any	Any	Any	Deny	"Catch-All" Rule - Everything not previously allowed is explicitly denied

Table 4.2: Sample Ruleset for Boundary Router

The main and internal firewalls would employ stateful inspection technology and could also include application-proxy capability, although this is not used in this example. The main firewall would perform the following actions:

- Allow external users to connect to the VPN server, where they would be authenticated.
- Pass internally bound SMTP connections and data to the proxy server, where the data can be filtered and delivered to destination systems.
- Route outbound HTTP traffic from the HTTP proxy and outbound SMTP traffic from the SMTP proxy.
- Subsequently deny other outbound HTTP and SMTP traffic.
- Subsequently allow other outbound traffic.

The internal firewall would accept inbound traffic from only the main firewall and the two application proxies. Furthermore, it would accept SMTP and HTTP traffic from the proxies only, not the main firewall. Lastly, it would permit all outbound connections from internal systems.

To make this example more applicable to a higher-security environment, several items could change, including the following:

- Internal and external DNS servers could be added to hide internal systems.

- PAT and NAT could be used to further hide internal systems.
- Outbound traffic from internal systems could be filtered, including possibly traffic to questionable sites or for services whose legality is questionable or because of management policies.
- Multiple firewalls could be employed for failsafe performance.

5. Firewall Administration

Given the sensitive role played by firewalls, the manner in which they are managed and maintained is critical.

5.1. Access To The Firewall Platform

The most common method for breaking into a firewall is to take advantage of the resources made available for the remote management of the firewall. This typically includes exploiting access to the operating system console or access to a graphic management interface.

For this reason, access to the operating system console and any graphic management interface must be carefully controlled. The most popular method for controlling access is through the use of encryption and/or strong user authentication and restricting access by IP address. Most graphic interfaces for firewall management incorporate some form of internal encryption. Those that do not can usually be secured using Secure Sockets Layer (SSL) encryption. Secure Sockets Layer will usually be an option for those graphic management interfaces that rely on the hypertext transport protocol (HTTP) for interface presentation. If neither internal encryption nor secure sockets layer are available, tunneling solutions such as the secure shell[18] (ssh) are usually appropriate.

For user authentication, several options exist. First, most firewall management interfaces incorporate some form of internal authentication. In many cases, this involves an individual userID and password that must be entered to gain access to the interface. In other cases, this can involve a single administration account and its corresponding password. In still other cases, some firewalls can support token-based authentication or other forms of strong authentication. These secondary forms of authentication typically encompass centralized authentication servers such as RADIUS and TACACS/TACACS+[19]. Both RADIUS and TACACS/TACACS+ provide external user accounting and authentication services to network infrastructure components and computer systems. RADIUS and TACACS/TACACS+ may also be integrated with token-based solutions to better enhance administration security.

5.2. Firewall Platform Operating System Builds

Another key factor in successful firewall environment management is platform consistency. Firewall platforms should be implemented on systems containing operating system builds that have been stripped down and hardened for security applications, i.e., a bastion host. Firewalls should never be placed on systems built with all possible installation options.

---

[18] ssh, short for Secure Shell, uses public key cryptography to authenticate connections between systems and encrypt the traffic. It is used often when SSL is not available or would not be appropriate. ssh can also tunnel other protocols, thus creating an authenticated connection for, as an example, FTP.

[19] RADIUS is short for Remote Authentication Dial-In User Service; TACAS is short for TAC Access Control Server. Both are userID and password authentication and accounting systems used by many Internet Service Providers (ISPs).

Firewall operating system builds should be based upon minimal feature sets. All unnecessary operating system features should be removed from the build prior to firewall implementation, especially compilers. All appropriate operating system patches should be applied before any installation of firewall components.

The operating system build should not rely strictly on modifications made by the firewall installation process. Firewall installation programs rely on a lowest common denominator approach; extraneous software packages or modules might not be removed or disabled during the installation process.

The hardening procedure used during installation should be tailored to the specific operating system undergoing hardening. Some often-overlooked issues include the following:

- Any unused networking protocols should be removed from the firewall operating system build. Unused networking protocols can potentially be used to bypass or damage the firewall environment. Finally, disabling unused protocols ensures that attacks on the firewall utilizing protocol encapsulation techniques will not be effective.

- Any unused network services or applications should be removed or disabled. Unused applications are often used to attack firewalls because many administrators neglect to implement default-restrictive firewall access controls. In addition, unused network services and applications are likely to run using default configurations, which are usually much less secure than production-ready application or service configurations.

- Any unused user or system accounts should be removed or disabled. This particular issue is operating system specific, since all operating systems vary in terms of which accounts are present by default as well as how accounts can be removed or disabled.

- Applying all relevant operating system patches is also critical. Since patches and hot fixes are normally released to address security-related issues, they should be integrated into the firewall build process. Patches should always be tested on a non-production system prior to rollout to any production systems. This pre-rollout testing should include several specific events:

    1. A change of the system time (minute-by-minute, and hour-by-hour).

    2. A change of the system date (both natural, and manual).

    3. Adding and deleting of appropriate system users and groups.

    4. Startup and shutdown of the operating system.

    5. Startup and shutdown of the firewall software itself.

    6. System backups, if appropriate.

- Unused physical network interfaces should be disabled or removed from the server chassis.

5.3. Firewall Failover Strategies

Many options exist for providing redundancy and failover services for firewall environments. These options range anywhere from using specially designed network switches to using customized "heartbeat" mechanisms to assess and coordinate the availability of the primary firewall so that a backup can take over in the event of a failure.

Network switches that provide load balancing and failover capabilities are the newest and most advanced solutions currently available. In a failover configuration, these switches monitor the responsiveness of the production firewall and shift all traffic over to a backup firewall in the event that there is a failure on the production system. The primary advantage to this type of solution is that the switch masquerades both firewalls behind the same MAC (Media Access Control – OSI Layer 2) address. This functionality allows seamless failover; in many cases, established sessions through the firewall are not impacted by a production system failure.

The heartbeat-based solutions typically involve a back-end or custom network interface to notify the backup system in the event of a primary system failure. These systems rely on established, reliable technology to handle failover. The primary drawback to this approach is that established sessions traversing the production firewall are almost always lost in the transition from production to backup resources.

The decision on which failover method to implement is often reduced to cost; the network switch-based failover solution is generally more expensive than a heartbeat-based system.

5.4. Firewall Logging Functionality

Nearly all firewall systems provide some sort of advanced logging functionality. As discussed previously, logging output from application-proxy gateway firewalls tend to be much more comprehensive than similar output from packet filter or stateful inspection packet filter firewalls. This is because application-proxy gateway firewalls are aware of a much larger portion of the OSI model.

The generally accepted common denominator for logging functionality is the UNIX syslog application. UNIX syslog provides for centralized logging, as well as for multiple options for examining and parsing logs. This logging program or daemon is available for nearly all major operating systems, including Windows® NT, Windows® 2000 and XP, and all UNIX and Linux variants.

Once a set of firewall logs has been passed to a centralized logging server, quite a few software packages are available to examine those logs (several are detailed in Appendix B). Syslog-based logging environments can also provide inputs to intrusion detection and forensic analysis packages.

Those firewalls that do not support any syslog interface must use their own internal logging functionality. Depending on the firewall platform, there are numerous third-party tools for log maintenance and parsing.

5.5. Security Incidents

There is no simple answer to the question: What is a security incident?

In general, a security incident is any event in which unauthorized individuals access or attempt to access computer systems or resources to which they do not have privileges. The severity of the incident can vary and it is up to individual companies or agencies to determine the exact definition of a security incident.

On the low end of the severity scale, a minor security incident might consist of basic network or system probes that are designed to map corporate or agency networks. If an unauthorized person executes these probes, a security incident has taken place. Due to the sheer volume of these types of events, most companies or agencies choose not to treat these events as security incidents.

At the middle of the severity scale, a security incident might take the form of active attempts to gain unauthorized access to a computer system or systems. At the high end of the severity scale is any successful attempt to gain unauthorized access to a system or resource. These events have the potential to interrupt production availability of resources and are therefore taken seriously. When identified, some organizations or agencies will attempt to prosecute the perpetrator or perpetrators. In all cases, the incidents should be reported[20].

In essence, the definition of a security incident will be determined by an organization's individual security policy.

During a security incident, the line administrators have several responsibilities. In an ideal world, restoration of production access can take place without impacting the forensic evidence necessary to prosecute an alleged perpetrator, but this is not always possible. Depending upon the security policy in effect at an organization or agency, system or security administrators might also have other responsibilities. In general, these responsibilities will be dictated by some management entity. These responsibilities should be delineated ahead of time.

Firewalls can provide a critical perspective in the context of a security incident – event correlation. The concept of event correlation involves the fact that firewalls are in a unique position in that nearly all network-based attacks must traverse a firewall in order to get into a network. This puts the firewall in the unique position of having oversight on unauthorized activities. For this reason, all firewalls and other logging systems, such as intrusion detection systems, should employ time synchronization. The most common mechanism for time synchronization is the network time protocol, or NTP. When all of the systems having oversight agree on the time, it is possible to reconstruct the phases of a security incident.

---

[20] Federal agencies must report security incidents to FedCIRC, the Federal Computer Incident Response Center, at http://www.fedcirc.gov.

5.6. Firewall Backups

The conduct and maintenance of backups are key points to any firewall administration policy. All firewalls should be subject to a Day Zero backup. All firewalls should be backed up immediately prior to production release.

As a general principal, all firewall backups should be full backups. There is no real requirement or need for incremental backups.

It is usually not possible to employ a centralized backup scheme due to the firewall's access control. Also, permitting access to a centralized backup server that is presumably located behind the firewall would present a high risk to the privacy of the backups. Therefore, most firewalls should be built with internal (or external) tape drives. There should never be tape medium present in the drive unless a backup is being performed.

It is also desirable (although not always possible) to deploy firewalls that have all critical filesystems burned to CDROM. For UNIX, this is more possible; the main filesystem requiring write access is the /var filesystem, and all system logs and spool directories can be found in this directory or filesystem. Deployment of Windows®-based firewalls with read-only filesystems is not possible at this time.

5.7. Function-Specific Firewalls

Very often, firewalls are implemented to protect certain special-purpose systems. While not perfect, a good example would be firewalls designed to protect telephone management systems. With the fairly recent rise of in-band PBX[21] management software, firewalls for this function have become important[22].

Traditionally, PBX resources have been managed using text terminals or proprietary management consoles. Within the last several years, however, it has become common for PBX vendors to include management software that requires Layer 3 in-band connectivity to manage the systems. This type of requirement is especially necessary for the newer generation of smaller, modular PBX systems. In fact, it is not at all uncommon for newer PBX systems to implement modularity through the use of Layer 3 network connections between PBX nodes.

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

- Always allow emergency (911) calls,

---

[21] Short for Private Branch Exchange, a private telephone network used within an organization.

[22] See NIST ITL Bulletin *Security for Private Branch Exchange Systems*, August 2000, and Special Publication 800-24, *PBX Vulnerability Analysis: Finding Holes in Your PBX Before Someone Else Does*, at http://csrc.nist.gov

- Disallow incoming modems,
- Disallow outgoing modems, and
- Allow all other traffic.

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

PBX firewalls provide an important complement to a network firewall, since one of the most overlooked vulnerabilities in organizations is dial-up access. Often, users configure their desktop PCs to allow modem access when the user is on travel or working from home. Even if the organization has a corporate policy against such modems, a significant percentage of users may violate that policy on occasion. Most remote access software does not provide strong identification and authentication, and users are often negligent in selecting strong passwords. The PBX firewall provides a central point of administration for telephone line security.

Placing a firewall to regulate access to PBX resources also creates an additional audit trail for access to the PBX resources. With a firewall in place, not only would the PBX be logging the management session, but the firewall would also provide such logs.

Appendix A. Terminology

The following definitions highlight important concepts used throughout this document:

Active Content

Active content refers to electronic documents that can carry out or trigger actions automatically on a computer platform without the intervention of a user. Active content technologies allow mobile code associated with a document to execute as the document is rendered.

Application Content Filtering

Application content filtering is performed by a software proxy agent to remove or quarantine viruses that may be contained in email attachments, to block specific MIME types, or to filter other active content such as Java™, JavaScript, and ActiveX® Controls.

Bastion Host

A bastion host is typically a firewall implemented on top of an operating system that has been specially configured and hardened to be resistant to attack.

Boundary Router

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

DMZ

Demilitarized Zone, a network created by connecting two firewalls. Systems that are externally accessible but need some protections are usually located on DMZ networks.

Extranet

An extranet is a virtual network created by connecting two intranets. An organization that connects remote locations with a VPN creates an extranet by linking its intranets together to form one virtual network.

Firewall Environment

A firewall environment is a collection of systems at a point on a network that together constitute a firewall implementation. A firewall environment could consist of one device or many devices such as several firewalls, intrusion detection systems, and proxy servers.

Firewall Platform

A firewall platform is the system device upon which a firewall is implemented. An example of a firewall platform is a commercial operating system running on a personal computer.

Firewall Ruleset

A firewall ruleset is a table of instructions that the firewall uses for determining how packets should be routed between its interfaces. In routers, the ruleset can be a file that the router examines from top to bottom when making routing decisions.

IDS

Intrusion Detection System, a software application that can be implemented on host operating systems or as network devices to monitor for signs of intruder activity and attacks.

Intranet

An intranet is a network internal to an organization but that runs the same protocols as the network external to the organization. Every organizational network that runs the TCP/IP protocol suite is an intranet.

IPSec

A standard consisting of IPv6 security features ported over to the current version of IP, IPv4. IPSec security features provide confidentiality, data integrity, and non-repudiation.

ISP

Internet Service Provider, an entity providing a network connection to the global Internet.

MIME

Multipurpose Internet Mail Extensions, an extensible mechanism for email. A variety of MIME types exist for sending content such as audio using the SMTP protocol.

NAT, PAT

Network Address Translation and Port Address Translation, used to hide internal system addresses from an external network by mapping internal addresses to external addresses, by mapping internal addresses to a single external address, or by using port numbers to link external system addresses with internal systems.

Proxy agent

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

SOHO

Small Office/Home Office, an acronym commonly used for classifying devices for use in small office and home office environments.

SSL

Secure Sockets Layer, based on public key cryptography, used to generate a cryptographic session that is private to a web server and a client browser.

VPN

Virtual Private Network, used to securely connect two networks or a network and a client system, over an insecure network such as the Internet. A VPN typically employs encryption to secure the connection.

Appendix B. Links and Resources

This appendix contains references to books and publications on Internet security and firewalls. There is also a section containing web links to sites with information on firewalls, threats and vulnerabilities, and related information[23]. This information is current as of the time of publication; readers are advised to consult the most up-to-date sources for firewall and Internet-related security.

B.1. NIST CSD Websites

The NIST Computer Security Division (CSD) maintains a website with information about its programs, copies of its publications (including this one), and information about many areas of computer security, including the following:

- Firewalls
- Intrusion detection
- Active content
- Viruses
- Threats and vulnerabilities
- General network security
- Policy creation and guidelines
- Risk analysis and assessment
- Training and education

This website can be accessed at http://csrc.nist.gov.

The CSD also maintains a related site with a database of threats and vulnerabilities and information about many public domain and vendor products. The site is particularly useful for administrators who need to know the vulnerabilities associated with their system configurations and which patches to apply. This website can be accessed directly at http://csrc.nist.gov/icat.

[23] This material is based on the MIS Training Institute "ISI Swiss Army Knife Reference." For more information, please see http://www.misti.com.

B.2. Books and Publications on Firewall Security

- Assembly Instructions Included (Cisco Routers); Gilbert Held; Network Magazine; January 2001
- Building A Floppy Firewall; Andreas Meyer; Sys Admin; January 2001
- Building Internet Firewalls – 2nd Edition; D. Brent Chapman & Elizabeth D. Zwicky; O'Reilly; 2000
- Building Linux and OpenBSD Firewalls; Wes Sonnenreich, Tom Yates; Wiley; 2000
- Cisco IOS: It's Not Just for Routing Anymore; Greg Shipley; Network Computing; May 31, 1999
- Cisco IOS 12 Network Security; Cisco Press/Macmillan Technical Publishing; 1999
- Cisco Security Architectures; Gil Held & Kent Hundley; McGraw-Hill; 1999
- Decipher Your Firewall Logs; Robert Graham; Internet Security Advisor; Mar/Apr 2000
- Firewall Configuration Done Right; Rik Farrow; Network Magazine; December 1998
- Firewall Vulnerabilities; Rik Farrow; Network Magazine; August 1999
- Firewalls 24Seven; Matthew Strebe, Charles Perkins; Sybex Network Press; 1999
- Firewalls Complete; Marcus Goncalves; McGraw-Hill; 1998 (includes CD-ROM with demo versions of major firewall products)
- Firewalls & Internet Security - Repelling the Wiley Hacker; Bill Cheswick & Steve Bellovin; Addison-Wesley; 1998
- FreeBSD Firewall Tools & Techniques; Michael Lucas; Sys Admin; June 2000
- Great Walls of Fire (Firewall Security); Linda Boyer; NetWare Connection; January 1997
- The 'Ins' and 'Outs; of Firewall Security; Mike Fratto; Network Computing; September 6, 1999
- Internet Firewalls & Network Security - Second Edition; Karanjit Siyan; New Riders Publishing; 1996
- Keeping Your Site Comfortably Secure: An Introduction to Internet Firewalls; NIST Special Publication 800-10
- Kicking Firewall Tires; Char Sample; Network Magazine; March 1998
- A Linux Internet Gateway; Marcel Gagne; Sys Admin; June 2000

- NAT: Network Address Translator; Ron McCarty; Sys Admin; March 2000
- Packet Filtering and Cisco's Way; Ron McCarty; Sys Admin; May 1999
- Router-Based Network Defense; Gilbert Held; Sys Admin; March 2000
- The Use of Routers in Firewall Setup; Matej Sustic; Sys Admin; May 2000

B.3. Books and Publications on Intrusion Detection & Incident Response

- Can You Survive A Computer Attack?; Rik Farrow & Richard Power; Network World; May 2000
- Deploying an Effective Intrusion Detection System; Ramon J. Hontanon; Network Magazine; 2000
- Detecting Intrusions Within Secured Networks; Dan Sullivan; Internet Security Advisor; Fall 1999
- FAQ: Network Intrusion Detection Systems; Robert Graham; www.robertgraham.com; March 2000
- Fcheck: A Solution to Host-Based Intrusion Detection; Ron McCarty; Sys Admin; December 2000
- An Introduction to Intrusion Detection and Assessment; Rebecca Bace; ICSA; 2000
- Intrusion Detection: An Introduction to Internet Surveillance, Correlation, Trace Back, Traps & Response; Edward G. Amoroso; Intrusion Net Books; 1998
- Intrusion Detection; Rebecca Bace; New Riders Publishing; 2000
- Intrusion Detection: Network Security Beyond the Firewall; Terry Escamilla; Wiley; 1998
- Intrusion Detection Primer; Benjamin J. Thomas; linuxsecurity.com; March 13, 2000
- Intrusion Detection Strategies & Design Considerations; Ron McCarty; Sys Admin; September 1999
- Investigating Potential Intrusions; Eric Maiwald; Internet Security Advisor; Fall 1999
- Snort – A Lock Inside an Intrusion Detection System; Kristy Westphal; Sys Admin; September 2000
- Watching the Watchers: Intrusion Detection; Greg Shipley; Network Computing; November 13, 2000

B.4. Websites – Firewall Security

- www.clark.net/pub/mjr/pubs/fwfaq (Marcus Ranum Firewall FAQ)
- www.firewall.com (numerous links to firewall references and software resources)
- www.nfr.com/forum/firewall-wizards.html (Firewall Wizards mailing list and archives)
- www.zeuros.co.uk (Rotherwick Firewall Resources)
- lists.gnac.net (GreatCircle Firewalls Digest mailing list and archives)
- www.cert.dfn.de/eng/fwl/ (German CERT firewall laboratory)
- www.nwconnection.com/ (Jan '97 issue - excellent technical tutorial on firewalls)
- www.robertgraham.com/pubs/ (several detailed white papers on firewalls and intrusion detection)
- www.cisco.com (Cisco Website – numerous how-to's FAQ on router security)
- www.phoneboy.com/fw1/ (Unofficial CheckPoint Firewall-1 FAQ & freeware site)
- www.icsa.net/ (International Computer Security Association – firewall certification)
- icat.nist.gov (ICAT vulnerability database, National Institute of Standards and Technology)
- www.sans.org/ (numerous documents and links to security sources)
- time.nist.gov (information on NTP)

Appendix C. Firewall Policy Recommendations

This appendix summarizes the recommendations contained in the main body of this document and adds other general recommendations. This appendix provides help to technical managers and policy writers in creating technically sound and maintainable policies that address the major security concerns and firewall issues.

C.1. General Recommendations

Organizations and agencies should use firewalls to secure their Internet connections and their connections to other networks. At remote locations, users should use personal firewalls and firewall appliances to secure their connections to the Internet and Internet Service Providers.

Organizations should view firewalls as their first line of defense from external threats; internal security must still be a top priority. Internal systems must be patched and configured in a timely manner.

Organizations must monitor incident response team reports and security websites for information about current attacks and vulnerabilities. The firewall policy should be updated as necessary. A formal process should be used for managing the addition and deletion of firewall rules.

Organizations should recognize that all system administration, especially firewall administration, requires significant time and training. Organizations should ensure that their administrators receive regular training so as to stay current with threats and vulnerabilities.

C.2. Recommendations for Firewall Selection

Organizations should examine carefully which firewall and firewall environment is best suited to their needs. Assistance is available from a number of commercial sites that deal with firewall selection and analysis; a list of evaluated products for use in U.S. federal agencies is maintained by the National Information Assurance Center at http://csrc.nist.gov/niap.

A firewall environment should be employed to perform the following general functions:

- Filter packets and protocols
- Perform Stateful inspection of connections
- Perform proxy operations on selected applications
- Log traffic allowed and denied by the firewall
- Provide authentication to users using a form of authentication that does not rely on static, reusable passwords that can be sniffed

The firewall should be able to filter packets based on the following characteristics:

- Protocol, e.g., IP, ICMP
- Source and destination IP addresses
- Source and destination ports (which identify the applications in use)
- Interface of the firewall that the packet entered

The proxy operations should, at a minimum, be operable on the content of SMTP, FTP, and HTTP protocol traffic.

Organizations and agencies may find that they need several firewalls to accomplish these items.

C.3. Recommendations for Firewall Environment

A boundary router or other firewall should be used at the Internet connection to create an external DMZ. Web servers and other publicly accessible servers should be placed on the DMZ so that they can be accessible as needed and still have some protections from the firewall. Internal users should be protected with an additional firewall.

Figure C.1 shows a general picture of a firewall environment. For remote users, a VPN is preferable. While a dial-in server could be located behind a firewall, it is more secure to combine it with a VPN server located at the firewall or external to the firewall so that remote connections can be securely authenticated and encrypted.

Intrusion detection is recommended as an additional safeguard against attacks. Figure C.1 shows network-based IDS; host-based IDS could be used on systems where high-speed throughput is not an issue, e.g., email servers.

Network address translation and split DNS are recommended to hide internal system names and addresses from external networks.

Remote users should use personal firewalls or firewall appliances when connecting to ISPs, regardless of whether dial-in or higher-speed connections are used.

C.4. Recommendations for Firewall Policy

A general risk assessment and a cost-benefits analysis should be performed on the network applications that the organization or agency has chosen to use. This analysis should result in a list of the network applications and the methods that will be used to secure the applications.

A firewall policy should be written to include a network applications matrix (or similar specification). This policy should be maintained and updated frequently as new attacks or vulnerabilities arise or as the organization's needs in terms of network applications change. This policy should make the process of creating the firewall ruleset less error-prone and more verifiable, since the ruleset can be compared to the applications matrix.

All firewall and security policies should be audited and verified at least quarterly.

The default policy for the firewall for handling inbound traffic should be to block all packets and connections unless the traffic type and connections have been specifically permitted. This approach is more secure than another approach used often: permit all connections and traffic by default and then block specific traffic and connections. No default policy for handling outbound traffic is included here; organizations should consider using outbound traffic filtering as a technique for further securing their networks and reducing the likelihood of internally based attacks.

As a general rule, any protocol and traffic that is not necessary, i.e., not used or needed by the organization and/or denied by policy, should be blocked via use of a boundary router and packet filtering technology. This will result in reduced risk of attack and will create a network environment that has less traffic and is thus easier to monitor.

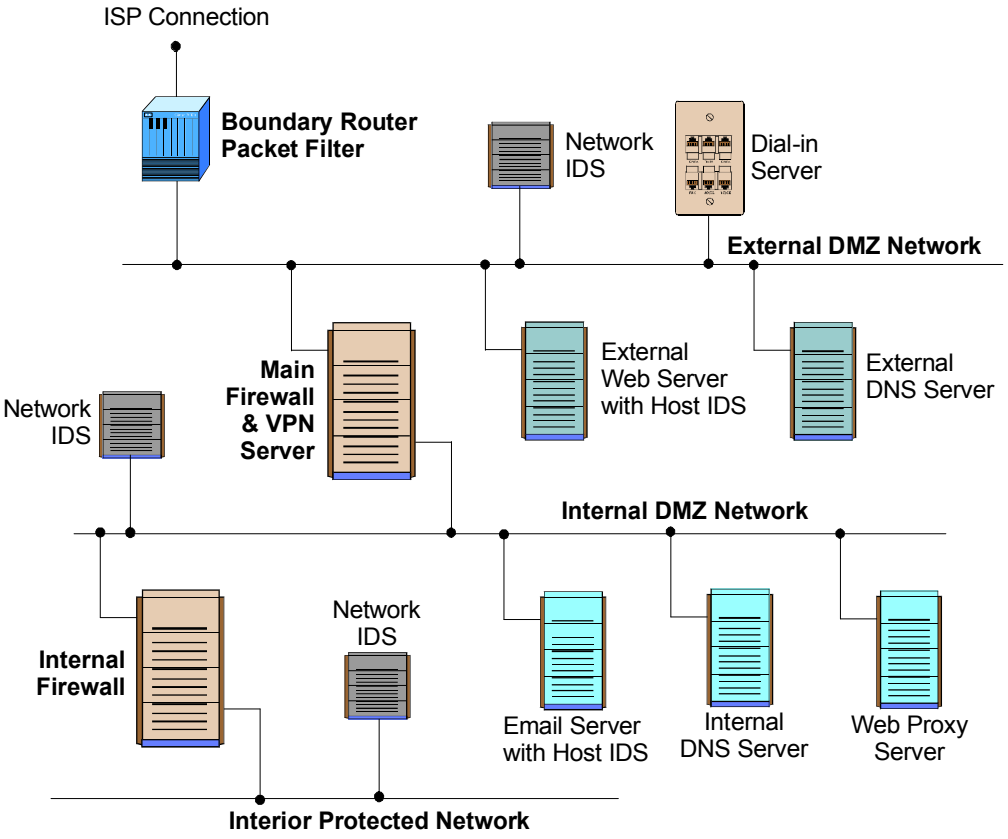

Figure C.1: Firewall Environment

Proxy applications should be used for out-bound HTTP connections and for in-bound/outbound email that are capable of the following operations:

- Blocking Java™ applets and applications
- ActiveX® and JavaScript filtering
- Blocking specific MIME extensions
- Scanning for viruses

Note: This is not a recommendation to *enable* blocking of active web content, but to be capable of blocking it should it be necessary. The decision to block active content, excluding viruses, should be weighed carefully, as blocking active content will render many websites unusable or difficult to use. Executable files in email attachments that could be blocked include the following:

.ade	**.cmd**	**.eml**	**.ins**	**.mdb**	**.mst**	**.reg**	**.url**	**.wsf**
.adp	**.com**	**.exe**	**.isp**	**.mde**	**.pcd**	**.scr**	**.vb**	**.wsh**
.bas	**.cpl**	**.hlp**	**.js**	**.msc**	**.pif**	**.sct**	**.vbe**	
.bat	**.crt**	**.hta**	**.jse**	**.msi**	**.pl**	**.scx**	**.vbs**	
.chm	**.dll**	**.inf**	**.lnk**	**.msp**	**.pot**	**.shs**	**.wsc**	

Organizations should not rely solely on the firewall proxies to remove the above content; web browsers should be set to appropriate security levels, and anti-virus software should be used on personal computers.

As stated previously, the overall policy of the firewall should be to block all inbound traffic unless that traffic is explicitly permitted. The following services and applications traffic thus should be blocked inbound by that policy, with exceptions noted[24]:

Application	Port Numbers	Action
Login services	telnet - 23/tcp	restrict w/ strong authentication
	SSH - 22/tcp	restrict to specific systems
	FTP - 21/tcp	restrict w/ strong authentication
	NetBIOS - 139/tcp	always block
	r services - 512/tcp - 514/tcp	always block
RPC and NFS	Portmap/rpcbind - 111/tcp/udp	always block
	NFS - 2049/tcp/udp	always block
	lockd - 4045/tcp/udp	always block
NetBIOS in Windows NT	135/tcp/udp	always block
	137/udp	always block
	138/udp	always block
	139/tcp	always block
	445/tcp/udp in Windows 2000	always block

[24] This policy is adapted from guidance from the CERT/CC (Computer Emergency Response Team/Coordination Center) and the SANS Institute. For more information, see http://www.cert.org/tech_tips/packet_filtering.html and http://www.sans.org/top20.htm.

Application	Port Numbers	Action
X Windows	6000/tcp - 6255/tcp	always block
Naming services	DNS - 53/udp	restrict to external DNS servers
	DNS zone transfers - 53/tcp	block unless external secondary
	LDAP – 389/tcp/udp	always block
Mail	SMTP - 25/tcp	block unless external mail relays
	POP - 109/tcp and 110/tcp	always block
	IMAP - 143/tcp	always block
Web	HTTP - 80/tcp and SSL 443/tcp	block unless to public Web servers
	may also want to block common high-order HTTP port choices - 8000/tcp, 8080/tcp, 8888/tcp, etc.	
"Small Services"	ports below 20/tcp/udp	always block
	time - 37/tcp/udp	always block
Miscellaneous	TFTP - 69/udp	always block
	finger - 79/tcp	always block
	NNTP – 119/tcp	always block
	NTP - 123/tcp	always block
	LPD - 515/tcp	always block
	syslog – 514/udp	always block
	SNMP - 161/tcp/udp, 162/tcp/udp	always block
	BGP - 179/tcp	always block
	SOCKS - 1080/tcp	always block
ICMP	block incoming echo request (ping and Windows traceroute)	
	block outgoing echo replies, time exceeded, and destination unreachable messages except "packet too big" messages (type 3, code 4). This item assumes that you are willing to forego the legitimate uses of ICMP echo request to block some known malicious uses.	

Table C.1: Summary of Ports/Protocols to Block

The following types of network traffic always should be blocked:

- Inbound traffic from a non-authenticated source system with a destination address of the firewall system itself.
- Inbound traffic with a source address indicating that the packet originated on a network behind the firewall.
- Inbound traffic from a system using a source address that falls within the address ranges set aside in RFC 1918 as being reserved for private networks.

- Inbound traffic from a non-authenticated source system containing SNMP (Simple Network Management Protocol) traffic.
- Inbound traffic containing IP Source Routing information.
- Inbound or outbound network traffic containing a source or destination address of 127.0.0.1 (localhost).
- Inbound or outbound network traffic containing a source or destination address of 0.0.0.0.
- Inbound or outbound traffic containing directed broadcast addresses.

C.5 Recommendations for Firewall Administration

If the firewall is implemented on a vendor operating system, (e.g., UNIX, Windows®) the operating system should be stripped of unnecessary applications and should be hardened against attack. All patches should be applied in a timely manner[25].

Firewall backups should be performed via an internally situated backup mechanism, e.g., tape drive. Firewall backups should not be written to any backup servers located on protected networks, as this may open a potential security hole to that network.

Firewalls should log activity, and firewall administrators should examine the logs daily. The Network Time Protocol (NTP) or another appropriate mechanism should be used to synchronize the logs with other logging systems such as intrusion detection.

An organization should be prepared to handle incidents that may be inevitable despite the protections afforded by the firewall environment. An incident response team should be created to assist the recovery from and analysis of any incidents[26].

---

[25] NIST's vulnerability database located at http://icat.nist.gov can be used to search for vulnerabilities associated with operating systems and applications, and to identify patches for correcting the vulnerabilities.

[26] The Federal Computer Incident Response Center (FedCIRC) is the central coordination facility for the topic of incident handling for civilian agencies of the federal government. See http://www.fedcirc.gov.

Appendix D. Index