

NIST Special Publication X-X

**Guide to Firewall
Selection and
Policy
Recommendations**

John Wack, Ken Cutler, Jamie Pole

NIST

**National Institute of
Standards and Technology**

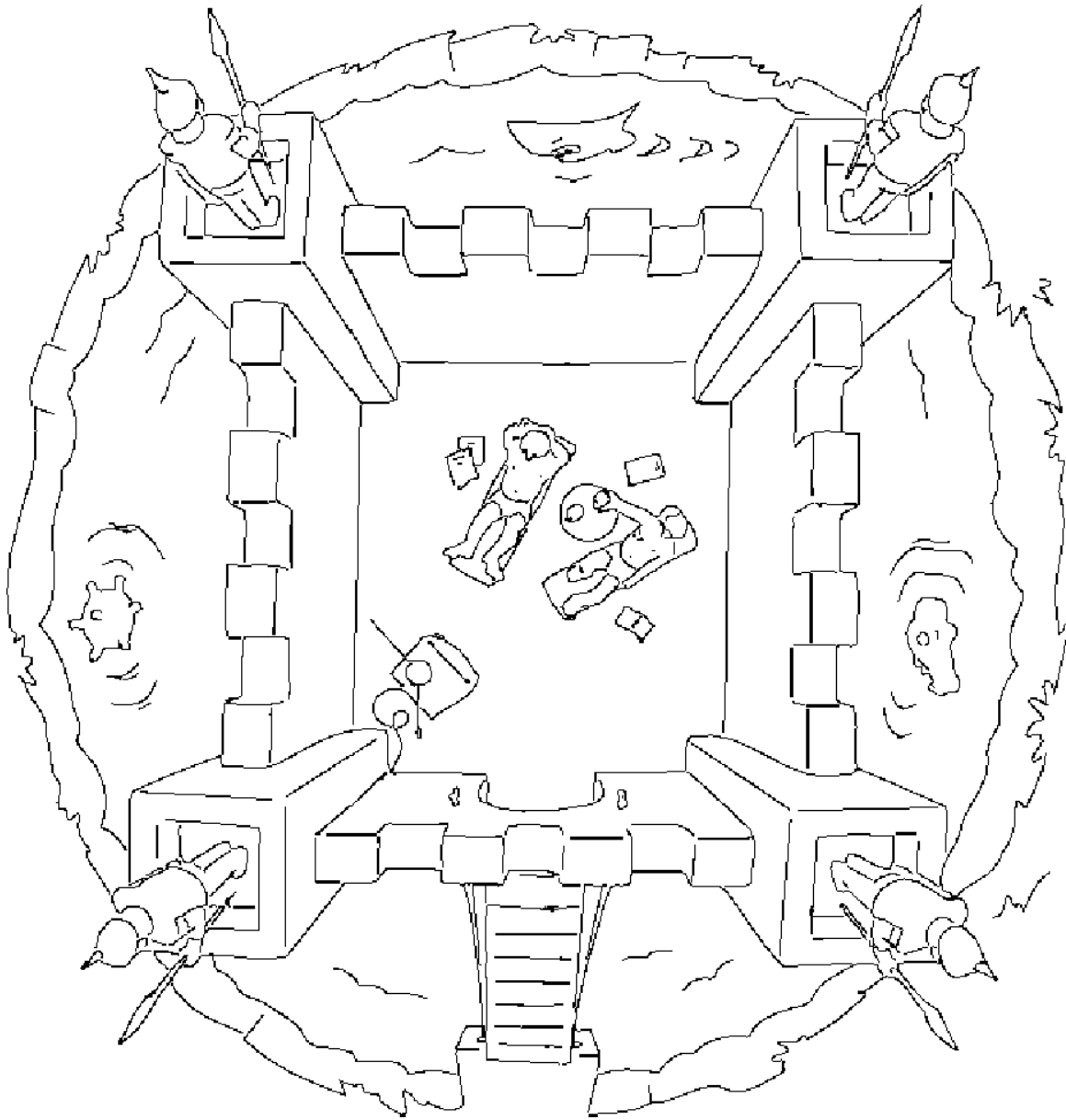
Technology Administration

U.S. Department of Commerce

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

DRAFT
October 01





Acknowledgements

The authors wish to express their thanks to staff at NIST who reviewed drafts of this document. In particular, Timothy Grance, Peter Mell, Gale Richter, and Murugiah Souppaya provided valuable insights that contributed substantially to the technical content of this document. NIST would also like to recognize Ken Cutler and Jamie Pole of MIS Training Institute, who authored substantial portions of this document under contract to NIST.

Table of Contents

1.	Introduction	1
1.1.	Document Purpose and Scope.....	1
1.2.	Audience and Assumptions.....	1
1.3.	Document Organization	2
2.	Overview of Firewall Platforms.....	3
2.1.	General Introduction to Firewall Technology.....	3
2.2.	Packet Filter Firewalls.....	6
2.3.	Stateful Inspection Firewalls.....	11
2.4.	Application-Proxy Gateway Firewalls.....	12
2.5.	Dedicated Proxy Servers	14
2.6.	Hybrid Firewall Technologies.....	16
2.7.	Network Address Translation.....	17
2.8.	Personal Firewalls/Personal Firewall Appliances.....	19
3.	Firewall Environments	23
3.1.	DMZ Networks	23
3.2.	Virtual Private Networks.....	25
3.3.	Intranets.....	26
3.4.	Extranets.....	27
3.5.	Infrastructure Components: Routers and Switches.....	27
3.6.	Intrusion Detection Systems.....	28
3.7.	Domain Name Service (DNS).....	31
4.	Firewall Security Policy	33
4.1.	Firewall Policy	33
4.2.	Implementing a Firewall Rulebase.....	35
4.3.	Testing Firewall Policy	37
4.4.	Firewall Implementation Approach	38
4.5.	Firewall Maintenance & Management.....	39
4.6.	Periodic review of information security policies	39
4.7.	A Medium-Security Sample Topology and Rulebase.....	40
5.	Firewall Administration	45
5.1.	Access to the firewall platform	45
5.2.	Firewall platform operating system builds.....	46
5.3.	Firewall Failover strategies	47
5.4.	Firewall logging functionality.....	47
5.5.	Security Incidents.....	48
5.6.	Firewall backups	49
5.7.	Function-Specific Firewalls	50
A.	Links and Resources.....	53
B.	Firewall Policy Recommendations.....	57

Table of Figures

Figure 2.1: OSI Communications Stack.....	3
Figure 2.2: OSI Layers Addressed by Modern Firewalls.....	4
Figure 2.3: OSI Layers Addressed by Packet Filters	6
Figure 2.4: Packet Filter used as Border Router	8
Figure 2.5: Sample Packet Filter Firewall Rulebase.....	9
Figure 2.6: OSI Layers Addressed by Stateful Inspection.....	11
Figure 2.7: Return Connection Rule	11
Figure 2.8: Stateful Firewall Connection State Table.....	12
Figure 2.9: OSI Layers Addressed by Application-Proxy Gateway Firewalls	13
Figure 2.10: Typical Proxy Agents	14
Figure 2.11: Application Proxy Configuration	16
Figure 2.12: Static Network Address Translation Table.....	18
Figure 2.13: Port Address Translation Table	19
Figure 3.1: A DMZ Firewall Environment	23
Figure 3.2: Service Leg DMZ Configuration to be Avoided	24
Figure 3.3: VPN Example	25
Figure 3.4: Extranet example	27
Figure 3.5: IDS Placement Throughout a Network.....	30
Figure 3.6: Split DNS example	32
Figure 4.1: Firewall Application Traffic Ruleset Matrix.....	34
Figure 4.2: Sample Medium-Security Firewall Topology	41
Figure 4.3: Sample Rulebase for Main Firewall	42
Figure 4.4: Sample Rulebase for Internal Firewall	43

1. Introduction

Firewall technology has improved substantially since it was introduced in the early 1990's, starting with simple packet-filtering firewalls to much larger firewalls capable of examining multiple levels of network activity and content, such as email and attachments that may contain viruses. 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 the neighborhood." Thus, firewall technology is now a standard part of any organization's network security architecture, and even home users on commercial dial-in connections routinely employ personal firewalls.

Modern firewalls are able to work in conjunction with each other and with other tools such as intrusion detection monitors. But firewalls do not provide complete protection from Internet-borne problems and therefore they are just one part of a total information security program. Firewalls are better viewed as the *last* line of defense for an organization; organizations must still make internal security a top priority. Firewall policy usually needs frequent if not daily updates, and this process is made easier if there is a strong information security policy already in place that can guide the much lower-level firewall policy.

1.1. Document Purpose and Scope

This document is intended to assist those responsible for network security by providing introductory information about firewalls and firewall policy. Non-technical management and those wishing to increase their knowledge of firewalls may find this document useful as well. 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 12, *Keeping Your Site Comfortably Secure: An Introduction to Firewall Technology*.¹

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. This document is not technically detailed, however it does assume some knowledge of TCP/IP, the

¹ Available at <http://csrc.nist.gov>.

protocol set used by the Internet, and various other aspects of networking and information security.

1.3. Document Organization

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. Chapter 3 describes various firewall environments, with suggestions for positioning firewalls and enabling them to work in conjunction with other firewalls. 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 would 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 suggests various recommendations for firewall administration.

Appendix A contains resources and links for more information about information security and firewalls in particular. Appendix B 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. The initial research into firewalls and related technologies took place in government subsidized projects, but nearly all recent advances in this field have taken place as a result of research conducted in the private sector.

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, but firewalls certainly have applicability in network environments that do not include or require Internet connectivity. For example, many corporate enterprise networks employ firewalls or firewall environments to restrict connectivity to and from 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.

There are several different types of firewall platforms currently available from vendors. The key differentiating factor among the various different firewall platforms is the amount of the Open Systems Interconnect (OSI) model that each given firewall platform is “aware” of and 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 will be addressed.

7	Application	7
6	Presentation	6
5	Session	5
4	Transport	4
3	Network	3
2	Data Link	2
1	Physical	1

Figure 2.1: OSI Communications Stack

A graphic depiction of the OSI model in Figure 2.1 shows a stack of networking layers. As a brief summary, the OSI model exists mainly to simplify the process of understanding how computer systems communicate on a network. Regardless of the operating systems and network media involved, the OSI model applies, however its implementation may or may not contain distinct layers – it is entirely implementation-independent. 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.

Moving up, 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 is the layer that identifies specific network applications and communication *sessions* as opposed to network addresses – a system may have any number of layer 4 connections with other systems on the same network. Terminology associated with the TCP/IP protocol suite includes the notion of *ports*, which are terminal constructs for connections: 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) represent end-user applications and systems and are not discussed here.

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

Layer	Name	Protocol example
7	Application	Email, Web browsers
4	Transport	TCP, session ID
3	Network	IP addressing
2	Data Link	Ethernet

Figure 2.2: OSI Layers Addressed by Modern Firewalls

Basic firewalls will be aware of a smaller number of layers; more advanced firewalls will cover a larger number of layers. In terms of functionality, firewalls capable of examining a large number of layers are capable of doing a more thorough job of firewalling. 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 understands only layers 2 and 3 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 are many add-on services that can exist. Some of these services include Network Address Translation (NAT), Dynamic Host Configuration Protocol (DHCP), encryption functionality such as Virtual Private Networks, and application content filtering. These services are discussed in subsequent paragraphs with the exception of NAT, which is discussed in Section 2.7.

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

Firewalls can also act as Virtual Private Network (VPN) endpoints. When firewalls are deployed as VPN endpoints, the organization or agency does not have to worry about passing unencrypted network traffic from systems behind the firewall to reach the VPN endpoint; the firewall encrypts the traffic and forwards it to the VPN endpoint. Most of the more popular firewalls nowadays incorporate this type of functionality. VPNs are discussed in greater detail in Section 3.2.

The final add-on discussed here involves content filtering technologies. This mechanism differs from the normal function of a firewall in that the firewall can also be made 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, JavaScript, and ActiveX². It should be noted, however, that firewall-based content filtering should not be relied upon as the sole content filtering mechanism for a organization or agency; it is possible to bypass these filters through the use of compression or encryption or other techniques. Once compression or encryption are introduced into the equation, the effectiveness of the filtering solution becomes a function of how well the platform can decompress or decrypt all traffic traversing the firewall and still maintain adequate quality of service.

² 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>

2.2. Packet Filter Firewalls

The most basic 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 rulebase. A sample packet filter firewall rulebase is included at the end of this section.

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 (its IP address).
- 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 (that system's IP address).
- The type of traffic, in other words, the specific network protocol being used to communicate between the source and destination systems or devices – possibly Ethernet.
- Possibly some characteristics of the Layer 4 communications sessions – the protocol such as TCP, and the source and destination ports of the sessions.
- 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 3 or more network interfaces.

7	Application	7
6	Presentation	6
5	Session	5
4	Transport	4
3	Network	3
2	Data Link	2
1	Physical	1

Figure 2.3: OSI Layers Addressed by Packet Filters

Packet filter firewalls are most 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 2 or more firewalls are employed to increase throughput or for fail-safe operations.

Some routers also permit firewall functionality at Layer 4, which constitutes an elementary form of *stateful inspection*, discussed more in Section 2.3. 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 Protocol³ (ICMP) – attackers have used this protocol to flood networks with traffic, thereby creating distributed denial-of-service (DDOS) attacks because the affected systems are overwhelmed with responding to the ICMP messages⁴. Packet filter firewalls also have the capability to block other attacks that take advantage of weaknesses in the TCP/IP suite.

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 shows a packet filter used as a border router. The router would accept packets from the untrusted network connection and perform access control according to the policy in place, e.g., block SNMP, permit HTTP, etc. It would then pass the packets to other 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 border router and the inner firewalls, sometimes referred to as the external DMZ network.

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

³ The ICMP protocol is at the same OSI layer as the IP protocol, and is used primarily for determining routing paths.

⁴ 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>

a packet filter firewall allows a given application, all functions available within that application will be permitted.

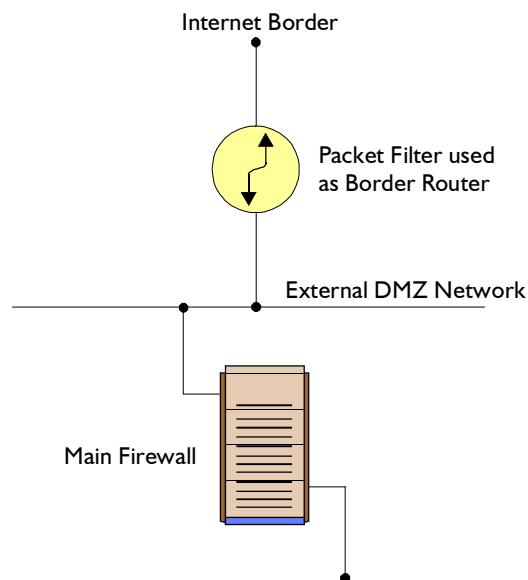


Figure 2.4: Packet Filter used as Border Router

- 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 awareness on the part of 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; preventative measures for these types of attacks are discussed in Chapter 3 of this document.
- Finally, due to the very small number of variables used in access control decisions, packet filter firewalls are very susceptible to security breaches caused by improper configurations. In other words, it is very 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.

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

With the current proliferation of advanced firewall features and functionality, it is difficult to identify a single firewall that contains only packet filter features. The closest available 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 failover⁵ solutions; several vendors offer hardware and software solutions for both high-availability and hot failover.

Source Address	Source Port	Destination Address	Destination Port	Action	Description
Any	Any	192.168.1.0	> 1023	Allow	Rule to allow return TCP Connections to internal subnet
192.168.1.1	Any	Any	Any	Deny	Prevent Firewall system itself from directly connecting to anything
Any	Any	192.168.1.1	Any	Deny	Prevent External users from directly accessing the Firewall system.
192.168.1.0	Any	Any	Any	Allow	Internal Users can access External servers
Any	Any	192.168.1.2	SMTP (25)	Allow	Allow External Users to send Email in
Any	Any	192.168.1.3	HTTP (80)	Allow	Allow External Users to access WWW server
Any	Any	Any	Any	Deny	"Catch-All" Rule - Everything not previously allowed is explicitly denied

Figure 2.5: Sample Packet Filter Firewall Rulebase

Figure 2.5 shows a sample of a packet filter firewall rulebase 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 rulebase 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, it would start at the top of the rulebase and work down through the rules – whenever it finds a rule that permits or denies the packet, it takes the appropriate action:

- *Accept*: the firewall passes the packet through the firewall as requested, subject to whatever logging capabilities may or may not be in place.

⁵ 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.

- *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 rulebase 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 previous actions, the “Discard” action may or may not generate log entries.

In Figure 2.5, 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 would indicate that an attacker is spoofing the firewall’s address, hoping that the firewall would pass this packet to an internal destination, which 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 (Hypertext Transport Protocol) data – email and web, respectively. The final rule 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.

It is important to note that if the last rule were accidentally skipped, all traffic originating from the outside would be permitted. When the rulebase is much longer and more detailed, mistakes can be made that could prove disastrous, therefore the rulebase should be examined very carefully before implementation.

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.

7	Application	7
6	Presentation	6
5	Session	5
4	Transport	4
3	Network	3
2	Data Link	2
1	Physical	1

Figure 2.6: OSI Layers Addressed by Stateful Inspection

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 may be 25 for SMTP, for example.

Packet filter firewalls must permit inbound network traffic on all of these “high-numbered” ports in order 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 unauthorized users who may employ a variety of techniques to abuse the expected conventions.

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

Figure 2.7: Return Connection Rule

Figure 2.7 shows the first line of the packet filter ruleset from Figure 2.5, 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.

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. Figure 2.8 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
3.223.212.212	1046	192.168.1.6	80	Established

Figure 2.8: 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.

Stateful inspection firewalls, being a superset of packet filter firewalls, are also capable of supporting network address translation in certain circumstances.

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) awareness.

The key factor separating application-proxy gateway firewalls from packet filter firewalls and stateful inspection packet filter firewalls is that application-proxy gateway firewalls

do not require a Layer 3 (Network layer) route between the inside and outside interfaces of the firewall – in fact, an application-proxy firewall could have just one network interface, although this is not recommended for performance reasons. But, in the event the application-proxy gateway software ceases to function, the firewall system has no ability to pass network packets through the firewall system. All network packets that traverse the firewall must do so under software (application-proxy) control.

Each individual application-proxy, also referred to as a proxy agent, interfaces directly with the firewall access control rulebase to determine whether a given piece of network

7	Application	7
6	Presentation	6
5	Session	5
4	Transport	4
3	Network	3
2	Data Link	2
1	Physical	1

Figure 2.9: OSI Layers Addressed by Application-Proxy Gateway Firewalls

traffic should be permitted to transit the firewall. In addition to the rulebase, each proxy agent has the ability to require authentication of each individual network user. This user authentication can take many forms, including, but not limited to the following:

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

Application-proxy gateway firewalls confer numerous advantages over packet filter firewalls and stateful inspection packet filter firewalls. To begin with, application-proxy gateway firewalls usually have good logging capabilities due to the firewall being able to examine the entire network packet, not simply 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 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 far 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.

Internal Networks	Proxy Agents	External Networks
	Telnet	
	FTP	
	NNTP	
	SMTP	
	DNS	
	HTTP	
	HTTPS	
	LDAP	
	Finger	

Figure 2.10: Typical Proxy Agents

The advanced functionality of application-proxy gateway firewalls also fosters several weaknesses 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 weakness is that application-proxy gateway firewalls tend to be limited in terms of support for new network applications and protocols, because 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, but those generic agents tend to negate many of the strengths of the application-proxy gateway architecture – 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 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 and determine which application is being targeted, and then hand off the traffic to the appropriate proxy server, e.g., an email or web proxy server. The proxy server would perform filtering or logging operations on the traffic and then forward it to internal systems (or another firewall). 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.

Dedicated proxies allow an organization to enforce user authentication requirements as well as other filtering and logging on any traffic that wishes to traverse 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 for years 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, this may place a heavier load on the firewall and those who administer it if the organization permits a wide range of network traffic to external sites.

In such a configuration, the dedicated proxy would be placed behind a firewall, which would only accept outbound user traffic from the dedicated proxy server. This configuration would force any user wishing to access external resources to use the dedicated proxy server, which could enforce user authentication and perform logging and filtering.

In addition to authentication and logging functionality, dedicated proxy servers are ideal 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
- Virus scanning and removal
- Macro virus scanning, filtering, and removal
- Application-specific commands, for example, blocking the HTTP “delete” command
- User-specific controls, including blocking certain content types for certain users

Figure 2.11 shows a sample diagram of a network employing a dedicated proxy server placed behind another firewall system:

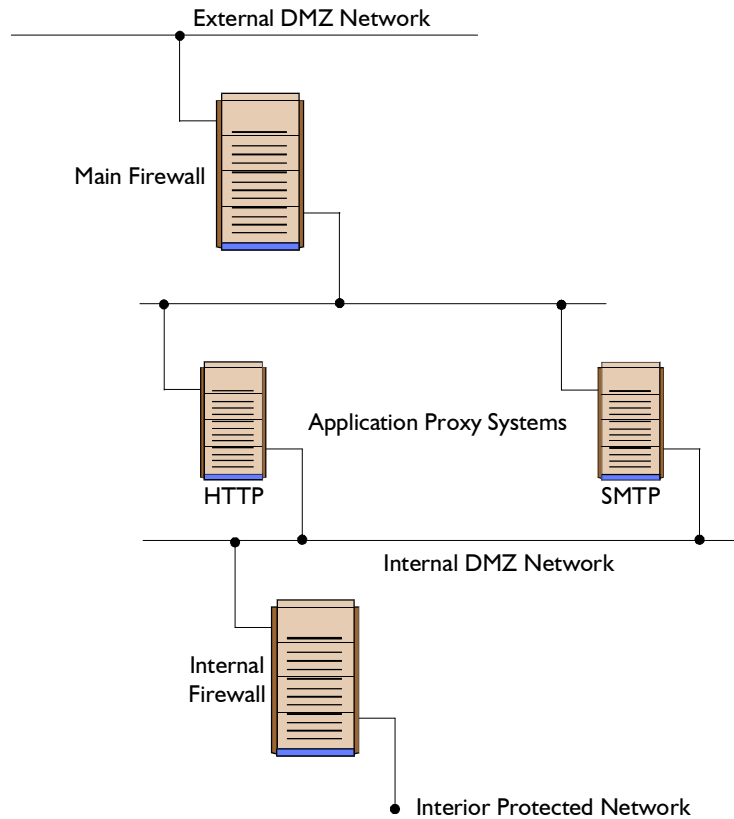


Figure 2.11: Application Proxy Configuration

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. The main result of these advances is that it is now common to see firewall products that 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 Protocol – Connectionless Transport) 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 extremely 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 extremely 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 their 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 1918⁶.

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 very seldom used, due to the scarcity of available IP address resources. With static network 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. Figure 2.12 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. In other words, 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 – it is not possible to make resources available to external users once they are placed behind a firewall that employs hiding network address translation – mapping in reverse from outside systems to internal systems isn’t possible, so 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

⁶ 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.

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

Figure 2.12: Static Network Address Translation Table

Port Address Translation (PAT): In a port address translation schema, the implementation is similar to hiding network address translation, with two primary differences. First, port address translation 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 is widely considered to be the state-of-the-art in terms of network address translation technology. PAT 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 Figure 2.13, 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

Figure 2.13: Port Address Translation Table

In terms of strengths and weaknesses, each type of network address translation has applicability in certain situations. The variable is 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.

2.8. 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. If a home user dials into an Internet Service Provider (ISP), they may have little firewall protections available to them from the ISP 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, the first being a software product that runs on the individual workstation. Such personal firewalls are installed on the system they are meant to protect; usually they 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 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 concentrator
- Network switch
- DHCP (Dynamic Host Configuration Protocol) server
- Network management (SNMP) agent
- 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. These devices 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 3.2.

When evaluating or choosing a personal firewall/personal firewall appliance solution, one of the most important issues is management of the device or application. 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 the exact same connectivity restrictions that 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 a 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 that user logs into. 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 for non-work related functions.

The ultimate solution is to use separate computers – for example, an organization could assign laptops to home users to be used for work functions only and that cannot be

connected to other networks (except the organization's) – even 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 using the computer for non-work functions. As well, if web security settings are set to reject certain types of content, this must remain in effect at all times. This also 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. The key point is that a personal firewall, like anti-viral software, cannot protect a system if it is disabled or reconfigured at certain intervals; it is an all or nothing proposition.

3. Firewall Environments

The concept of a firewall environment is very simple. A firewall environment consists of a set of systems and components that are involved in providing or supporting the complete firewall functionality at a given point on a network. This environment, in a simple 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. DMZ Networks

The most common firewall environment implementation is known as a demilitarized zone or DMZ network. These DMZ networks exist between firewalls and other access control points.

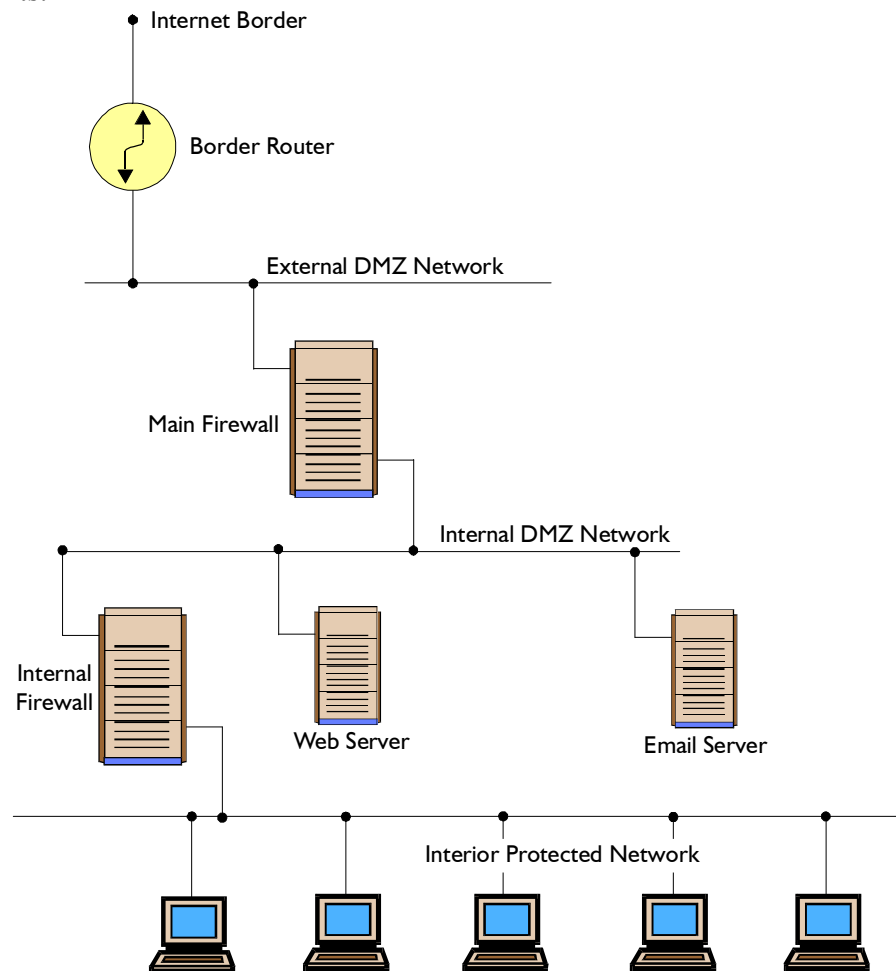


Figure 3.1: A DMZ Firewall Environment

DMZ networks serve as attachment points for computer systems and resources that are not entirely trusted. In other words, a computer system or resource that provides information or services to anonymous users should be placed in a DMZ network. DMZ networks are also very useful for securing resources shared between organizations or other entities. For example, many financial market data providers have implemented large DMZ networks for use in distributing their data to clients without giving those clients access to sensitive resources inside the provider’s networks. The DMZ networks allow market data providers or other organizations to make resources available in a fashion that prevents unauthorized dissemination of privileged or otherwise valuable information.

DMZ networks are typically implemented as network switches or concentrators 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.

One DMZ network configuration that should be avoided 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, one network interface attaches to an internal connection point, such as a network concentrator or network switch, and the third network interface forms the DMZ network. The major problem with this configuration is that it subjects the firewall to an increased risk of service degradation during a denial-of-service (DOS) attack. 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.

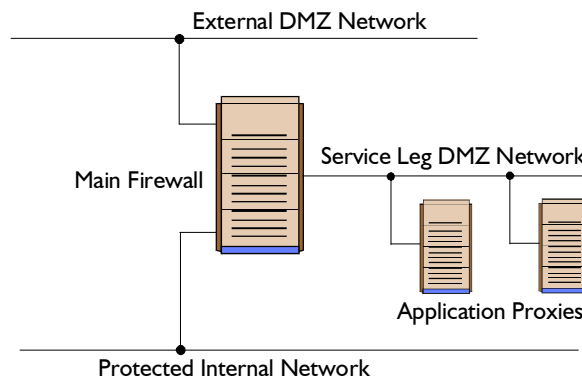


Figure 3.2: Service Leg DMZ Configuration to be Avoided

3.2. Virtual Private Networks

Another extremely valuable use for firewalls and firewall environments is the construction of Virtual Private Networks (VPNs). The concept of a virtual private network is very simple – a virtual network is constructed on top of existing network media. This virtual network can also be encrypted, but such encryption is not strictly necessary.

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 finds increasing use in the area of providing remote user access to corporate 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. Using virtual private network technology, a 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, so this mechanism is considered to be extremely cost-effective.

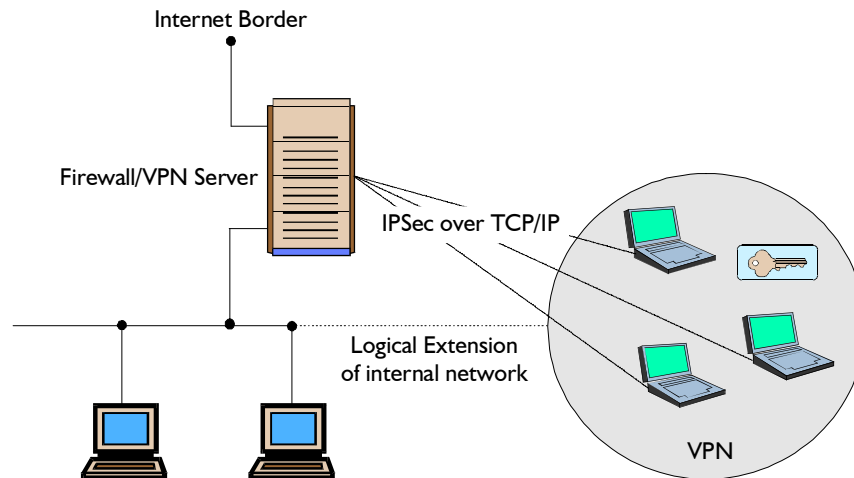


Figure 3.3: VPN Example

Virtual private network technology can also be used to create secure networks between organizations or agencies. For example, many financial market data providers now offer the option of information delivery via virtual private networks over the global Internet.

On the protocol level, there are several possible choices for a modern virtual private network. The first, and perhaps the most common at the moment is a set of protocols known as IPSec⁷. The IPSec standards consist of IPv6 security features ported over to IPv4, the version of IP in use today on the Internet. For the most part, IPSec is currently

⁷ See NIST ITL Bulletin *An Introduction to IPSec*, March 2001, at <http://csrc.nist.gov>

used strictly for providing VPN services, but IPSec does, in fact, contain additional functionality. Other current VPN standards include such protocols as PPTP (Point-to-Point Tunneling Protocol), a Microsoft standard, and the L2TP (Layer 2 Tunneling Protocol), which has been advanced by the open source movement.

There are several schools of thought regarding which device is most suitable for termination of virtual private network circuits. In most cases, the firewall is the best candidate for this function. Accordingly, most firewall vendors offer virtual private network clients that allow for remote user access to corporate networks. Several vendors also offer virtual private network solutions that do not involve the firewall, but these solutions are generally considered less secure than those involving the firewall. Figure 3.3 shows a VPN that is terminated by the firewall and that provides 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 laptops and the internal network.

It is also extremely important to understand that advanced virtual private network functionality does not come without a price. For example, if you choose to encrypt your virtual private network traffic, you should be aware that there will be a decrease in performance commensurate with (a) the amount of traffic flowing across your virtual private network, and (b) the type/length of encryption being used. Performing encryption in hardware will significantly increase performance, however, and should be considered. For some DMZ environments, the added traffic associated with virtual private networks might require additional capacity planning and resources.

3.3. Intranets

Nearly any discussion of enterprise network infrastructures ends up involving the concept of an intranet.

Despite the deference under which intranets are discussed, the definition of an intranet is actually very simple: 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 3.1, the internal protected network is an example of an intranet configuration.

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.

Given that 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.

3.4. Extranets

After Intranets and the Internet, Extranets form the third piece of the modern enterprise connectivity picture. An extranet is usually a business-to-business intranet, that is, two intranets 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, with the exception being 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 without permitting remote users to access an intranet. Extranets employ TCP/IP protocols, along with the same standard applications and services.

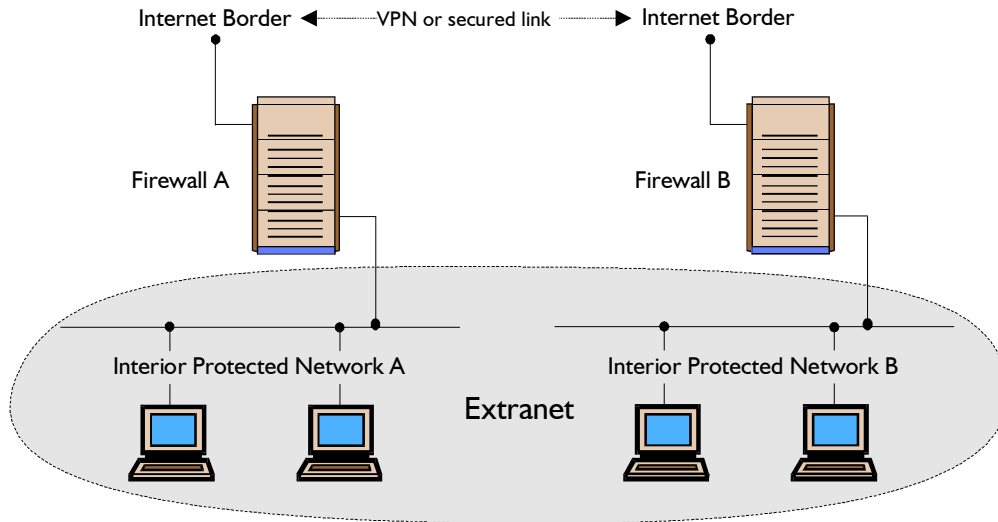


Figure 3.4: Extranet example

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.

3.5. Infrastructure Components: Routers and Switches

The effectiveness of a firewall environment is a function of many different factors. One of these factors is the type and quality of the infrastructure components utilized within the environment.

Infrastructure components are as varied as the networks they build. Components range from routers to network concentrators and network switches.

Routers are Layer 3 devices, and are the precursors to packet filtering firewalls. In fact, as stated earlier, packet filter firewalls are nothing more than routers with access control lists. Given that nearly all routers support some type of access control functionality, routers are the perfect complement to firewalls. In fact, the generally accepted design philosophy is that boundary routers should protect firewall devices before the firewall devices ever have to protect themselves. This principle ensures that the boundary router is able to compensate for any operating system or platform specific vulnerabilities that might be present on the firewall platform.

Above and beyond routers and firewalls, there are infrastructure devices that exist to provide connectivity between systems. The most simple of these connection devices is the network concentrator. Network concentrators are devices that function at Layer 1 of the OSI model. In other words, there is no real intelligence in network concentrators. Network concentrators exist only to provide physical attachment points for networked systems or resources. There are numerous weaknesses associated with network concentrators. First and foremost, network concentrators allow any device connected to them to see the network traffic destined for, or originating from any other device connected to that same network concentrator. For this reason, network concentrators should not be used to build DMZ networks or firewall environments.

The most advanced infrastructure device currently available 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 ideal infrastructure devices on which to implement DMZ networks and firewall environments.

3.6. Intrusion Detection Systems

Another modern development in network security is the concept of Intrusion Detection Systems (IDS)⁸. In essence, intrusion detection systems 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. This reactive capability is among the most advanced security measures currently available.

⁸ See NIST Special Publication 800-31, *Intrusion Detection Systems*, at <http://csrc.nist.gov>

There are currently two different types of intrusion detection systems 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 have a very high level of granularity in terms of the types of threats they can detect. In terms of weakness, there are several issues with host-based intrusion detection:

- 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. For example, it is not at all uncommon for UNIX host-based intrusion detection systems to degrade systems to the point that they require periodic reboots subsequent to installation of the intrusion detection system.

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 all 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 nearly always 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. There are also several issues with network-based intrusion detection:

- 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 fact alone makes it possible to bypass most network-based intrusion detection systems.
- 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 concentrators for network attachment points, especially in DMZ networks), network-based intrusion detection systems cannot function without special switch configurations (port mirroring, etc.). Many network switches, especially the less costly ones, lack such functionality.
- 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. The key concept is that 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.

The caveat with all existing types of intrusion detection is that they are not difficult to bypass if the attacker is knowledgeable. This fact must be made very clear during any intrusion detection system implementation. The second caveat is that intrusion detection systems generate voluminous logs, which must be examined very carefully if the intrusion detection system is to be effective. One of the biggest single issues with intrusion detection system implementation is the handling of false-positive notifications. Automated systems are prone to mistakes, and human differentiation of possible attacks is very resource-intensive.

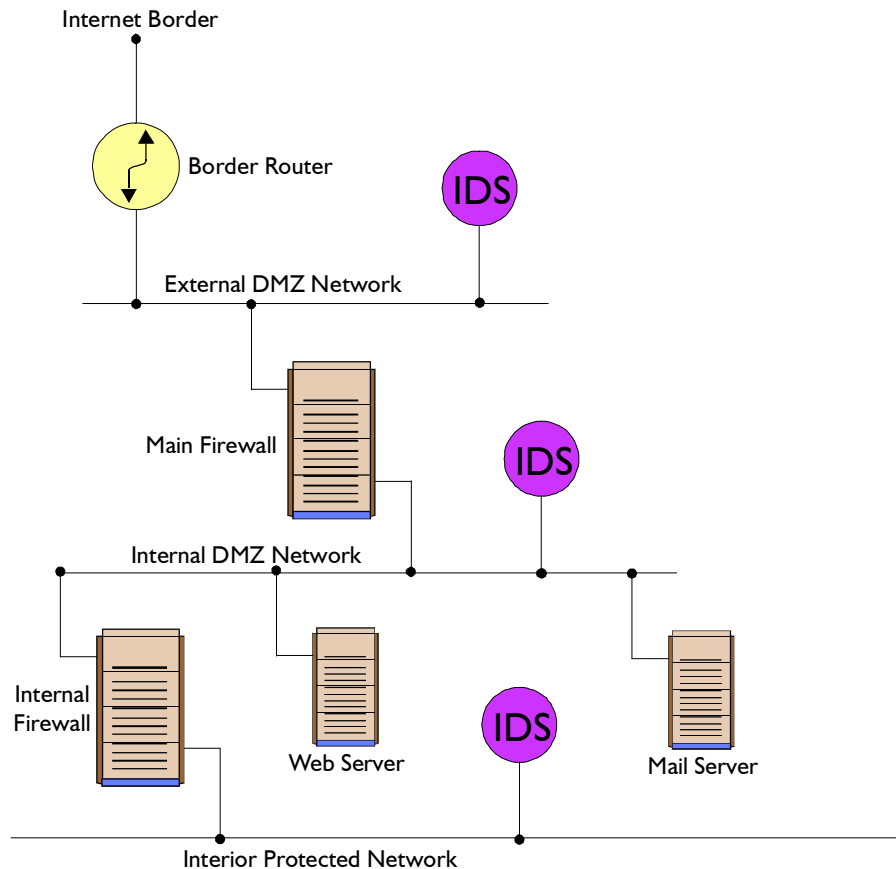


Figure 3.5: IDS Placement Throughout a Network

To properly implement an intrusion detection system solution, an organization must have a thorough understanding of the flow of data across their networks and systems. 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, an organization stands a much better chance of “noticing” a security incident in-progress. From a network-based intrusion detection standpoint, it is important to place intrusion detection systems at any location where 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.

3.7. Domain Name Service (DNS)

The domain name service (DNS) is critical to most corporate enterprise networks, and is certainly 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, possibly excepting 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. Best practice is that an organization maintains 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 such that it would resolve (find) names for internal systems, such 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 DMZ, but not the internal network. In other words, these systems only would be visible to the rest of the Internet.

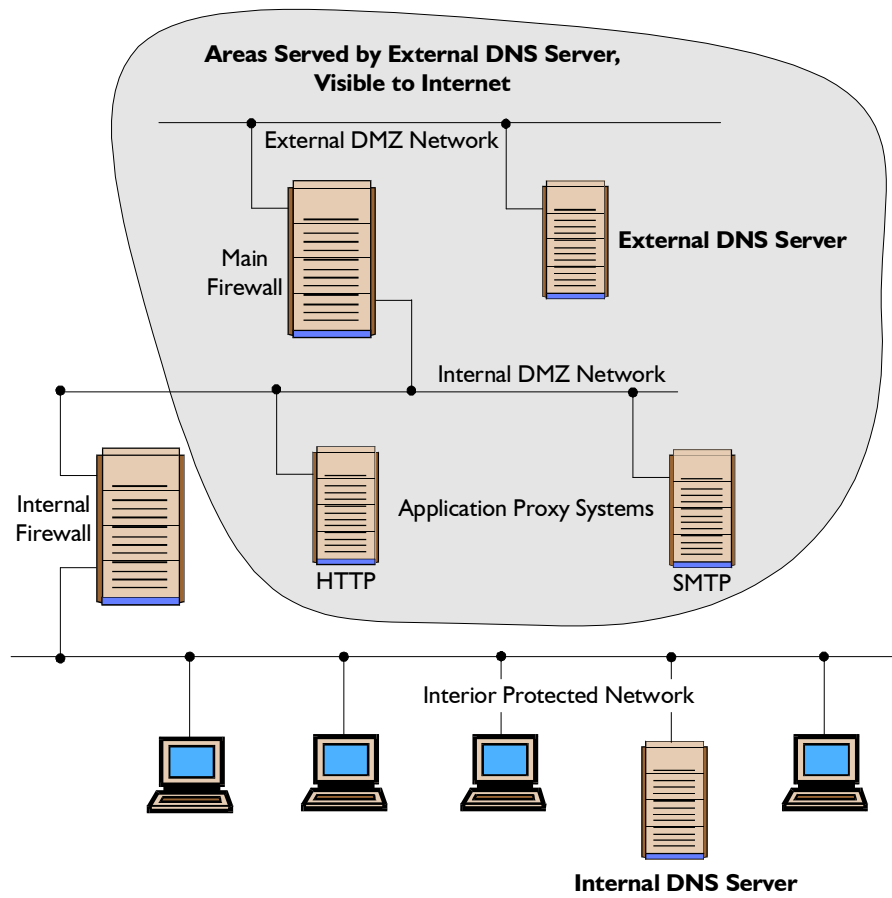


Figure 3.6: Split DNS example

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, however 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, therefore without a policy to guide firewall implementation and administration, the firewall itself may become a security problem. This section presents several steps for achieving a firewall policy and then presents 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 will handle applications traffic such as web, email, or telnet. Additionally, the policy will describe how the firewall is to be managed and updated. Finally, the policy will contain the steps involved in updating the policy itself.

Before a firewall policy can be created, some form of risk analysis must be performed on the applications that have been deemed 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 of creating this list is not detailed here⁹, 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.

The result of the cost-benefits 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 Figure 4.1.

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

1. Identification of network applications deemed necessary,
2. Identification of vulnerabilities associated with applications,

⁹ 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>

3. Cost-benefits analysis of methods for securing the applications,
4. Creation of applications traffic matrix showing protection method, and
5. Creation of firewall rulebase based on applications traffic matrix.

TCP/IP APPLICATION SERVICE	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); ftpusers feature	Permit	Application Proxy with User Authentication
"	Any	PC - TCP/IP	Client Only; Anti-Virus	Permit	Application Proxy with User Authentication
TFTP	Any	Unix Server with Diskless Clients Only	Secure Mode; Permit tftp to Limited Directories	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 Authentication
"	Any	PC - TCP/IP	Client Only	Permit	Application Proxy with User Authentication
"	Any	Router/Firewall	2 Password Layers; Token Authentication	Token Authentication	Reject
NFS	Any	UNIX	Limit Exports; Host/Groups (Granular Access)	Reject All, except by Written Authorization	Reject
"	Any	PC - TCP/IP	Client Only	Reject	Reject
NetBIOS over TCP/IP	Any	Windows NT/95/WFW	Limit Access to Shares	Permit Local Domain Only; Reject Others	Reject

Figure 4.1: Firewall Application Traffic Ruleset Matrix

4.2. Implementing a Firewall Rulebase

Most firewall platforms utilize rulebases as their mechanism for implementing security controls. The contents of these rulebases determine the actual functionality of a firewall. Depending on the firewall platform architecture, firewall rulebases can contain various pieces of information. Nearly all rulebases, however, will contain the following fields, as a minimum:

- The source address of a piece of network traffic, i.e., its source IP address.
- The destination address of a piece of network traffic, i.e., the destination IP address.
- The protocol, e.g., IP, ICMP, or TCP.
- The type of data contained in a piece of network traffic, such as the protocol and the source and destination ports. This characterization will nearly always involve a specific network application, such as Telnet, or SMTP.

The key point with firewall rulebases is that the greater the number of possible rulebase fields, the more advanced the firewall. It should also be noted that firewall rulebases tend to become increasingly complicated with age. For example, a new firewall rulebase might contain entries to accommodate only outbound user traffic and inbound email traffic (along with allowing the return inbound connections required by TCP/IP), but that same firewall rulebase 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.

After completing the applications traffic matrix, one can start to assemble the firewall rulebase. 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 rulebases should be built to be as specific as possible with regards to the network traffic they control. A key principle in firewall design is that rulebases 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.

When assembling the rulebase, there are certain types of network traffic that should *always* be blocked. Without utilizing vendor-specific terminology, these traffic types include the following:

- *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.* Given the fact that ICMP can be used to map the networks behind certain types of firewalls, ICMP should never be passed in from the Internet, or from any untrusted external network.
- *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.* For reference purposes, RFC 1918 reserves the following address ranges for private networks:
 - 10.0.0.0 – 10.255.255.255 (Class A, or “/8” in CIDR¹⁰ notation)
 - 172.16.0.0 – 172.31.255.255 (Class B, or “/12” in CIDR notation)
 - 192.168.0.0 – 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 rulebase 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 very 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 very rarely used, and valid applications are even less common on the Internet.

¹⁰ 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.

- *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 traffic containing directed broadcast addresses.* A directed broadcast is often used to initiate a broadcast propagation attack such as SMURF¹¹. Directed 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 rulebase 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 anything from sending email notification, to paging appropriate personnel.

4.3. Testing Firewall Policy

Policies are implemented every day. It is far more rare that any of these policies are actually 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. Many organizations utilize this type of review, and it is especially well suited to those organizations that do not employ in-house security and firewall expertise.

¹¹ 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>

The second methodology involves actual in-place configuration testing. In this methodology, the organization actually utilizes tools that assess the configuration of a device by attempting to perform operations that should be prohibited. This type of review is very well suited to those companies or agencies that employ internal firewall and security expertise. Fewer organizations employ this type of review, due to the costs associated with maintaining this type of specialization in-house. Another reason that fewer organizations employ this type of review is the potential cost of the tools required to complete the review. Although these reviews can be completed with public-domain tools, many companies, especially those subject to some type of regulatory requirements, will choose to employ costly commercial tools.

While the second methodology is more rigorous, both methodologies ideally should be employed.

Regardless of the type of review chosen, 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 very important that the firewall system itself be examined 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, one of the most basic decisions required is whether the firewall should be implemented as an appliance, or on top of a commercial operating system. While this decision will be largely determined by organization or agency requirements, there are some factors that can be used to differentiate between the two options.

First, it must be understood that in very general terms, appliance-based firewalls should be more secure than those implemented on top of commercial operating systems. The reason for this is one of simplicity – 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 to implementing firewalls on top of commercial operating systems is scalability. If an environment requires improved performance, it is usually a simple matter to 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 to 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 system¹².

This decision must be made based on relative costs, as well as perceptions 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 configuration. With a command-line interface, the administrator is presented a command prompt, and he or she then configures the firewall by typing commands into that command prompt. This technique tends to be error-prone, given the fact that the administrator's typing ability becomes a variable for firewall security posture. The primary advantage to command-line configuration is that given a skilled and experienced administrator, firewall configuration can usually be accomplished much more quickly than with a graphic interface.

The second (and most common) mechanism is through a graphic user interface. Graphic interfaces have the advantage of simplicity. Using a graphic interface, even novice administrators can 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.

Regardless of which option is chosen (or available), great care must be taken to ensure that all network traffic dealing with firewall system management be secured. For web-based interfaces, this security will likely be implemented through Secure Sockets Layer¹³ (SSL) encryption, along with a User ID and 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.

4.6. 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

¹² 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>

¹³ 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 are for secure financial transactions in which credit card information must be kept private from potential third-party observers of communications traffic.

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 such events as the implementation of major enterprise computing environment modifications, as well as any occurrence of a major information security incident.

Along with review of information policy, it is extremely important that in-place firewall installations as well as systems and other resources are audited on a regular, periodic basis. In some cases, these periodic reviews can be conducted “on paper”, in other words, utilizing hardcopy configurations provided by appropriate systems administration staff. In other cases, periodic reviews should involve actual audits and analyses of production and backup infrastructure components, computer systems and other various types of resources.

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 involves 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. Another circumstance in which a seeded penetration might be employed is when a organization or agency wants to limit the scope of an analysis to a given environment or set of systems.

A blind penetration involves 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.

4.7. A Medium-Security Sample Topology and Rulebase

This section presents a sample firewall topology and rulebase that is based on a medium-risk environment. The definition of a medium-risk environment is necessarily vague, however it would include the following:

- All internal network traffic permitted outbound to all sites

- Inbound SMTP (email) permitted to the firewall where it is passed to a proxy server
- Inbound HTTP (web) traffic permitted to the firewall where it is passed to a proxy server
- Inbound connections from remote systems permitted to the firewall’s VPN port where it is passed to internal systems
- All other inbound traffic blocked

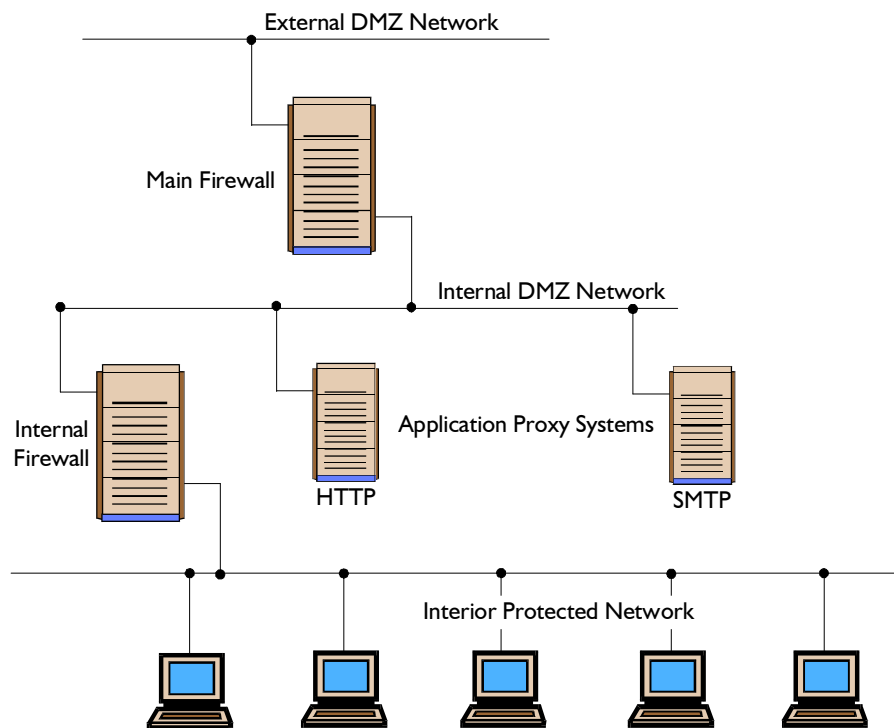


Figure 4.2: Sample Medium-Security Firewall Topology

In this example, the HTTP application proxy would cache certain inbound traffic for performance reasons, and it could also filter active content such as Java, JavaScript, or ActiveX controls. The SMTP application proxy would examine all email attachments or in-line content for viruses and quarantine the infected code as necessary.

The sample topology for this network is shown in Figure 4.2. An external DMZ network would connect to the Internet either directly or with a packet filter serving as a border router – Section 2.2 detailed reasons why a packet filter might be preferable. The firewall would incorporate a VPN port for remote users; such users would need VPN client software to connect to the firewall. Email and web traffic inbound would connect

to the firewall first, which would pass it on to application proxy servers located on an internal DMZ.

A sample rulebase for the Main Firewall would look as follows, in Figure 4.3. It contains the default blocking rules described as in Section 4.2. Note: this rulebase is greatly simplified; a real example would involve vendor-specific conventions and other details.

Source Address	Source Port	Destination Address	Destination Port	Action	Description
Any	Any	192.168.1.0	> 1023	Allow	Rule to allow return TCP Connections to internal subnet
192.168.1.1	Any	Any	Any	Deny	Prevent Firewall system itself from directly connecting to anything
Any	Any	192.168.1.1	VPN	Allow	Allow External users to connect to VPN server
Any	Any	192.168.1.2	SMTP (25)	Allow	Allow External Users to send Email to proxy
Any	Any	192.168.1.3	HTTP (80)	Allow	Send inbound HTTP to proxy
Any	Any	192.168.1.1	Any	Deny	Prevent External users from directly accessing the Firewall system.
192.168.1.0	Any	Any	Any	Allow	Internal Users can access External servers
Any	Any	Any	Any	Deny	"Catch-All" Rule - Everything not previously allowed is explicitly denied

Figure 4.3: Sample Rulebase for Main Firewall

Rule 1 allows return packets from established connections to return to the source systems. Rule 3 permits inbound connections to the main firewall’s VPN port, rules 4 and 5 tell the firewall to pass SMTP and HTTP traffic to the respective application proxies, and rule 6 then denies all other inbound connections to the firewall.

The internal firewall also needs a rulebase. In this example, it would accept inbound traffic from only locations: the main firewall and the two application proxies. Furthermore, it would accept SMTP and HTTP traffic from the proxies only. Lastly, it would permit all outbound connections from internal systems. A sample rulebase for the internal firewall is shown in Figure 4.4.

Source Address	Source Port	Destination Address	Destination Port	Action	Description
192.168.1.2	SMTP	Any	Any	Allow	Allow traffic from SMTP proxy
192.168.1.3	HTTP	Any	Any	Allow	Allow traffic from HTTP proxy
192.168.1.1	SMTP	Any	Any	Deny	Deny SMTP traffic from main firewall
192.168.1.1	HTTP	Any	Any	Deny	Deny HTTP traffic from main firewall
192.168.1.1	Any	Any	Any	Permit	Allow all other traffic from main firewall
192.168.1.0	Any	Any	Any	Allow	Internal Users can access External servers
Any	Any	Any	Any	Deny	"Catch-All" Rule - Everything not previously allowed is explicitly denied

Figure 4.4: Sample Rulebase for Internal Firewall

Rules 1 and 2 permit SMTP and HTTP traffic to pass from the application proxies to internal systems; rules 3 and 4 deny any such traffic from the main firewall. Rule 5 permits other traffic from the main firewall to pass to internal systems. Rule 6 permits traffic from internal systems to external systems, and rule 7 denies any other traffic not expressly permitted.

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 pornographic sites or for services whose legality is under question or simply because the management has deemed that as claims on staff productivity.
- Multiple firewalls could be employed for failsafe performance.

5. Firewall Administration

Given the extremely 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 shell¹⁴ (ssh) are usually appropriate.

For user authentication, there are several options. 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+¹⁵. 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.

¹⁴ 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.

¹⁵ 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).

5.2. Firewall platform operating system builds

Another key factor in successful firewall environment management is platform consistency. It is extremely important that firewall platforms be implemented on systems containing operating system builds that have been stripped down and hardened for security applications. Firewalls should never be built on top of “kitchen sink” builds, or builds which contain all possible installation options.

Firewall operating system builds should be based upon minimal feature sets – it is extremely important that all unnecessary operating system features are removed from the build prior to firewall implementation, especially compilers. Likewise, it is very important that any appropriate operating system patches are applied before any installation of firewall components.

It is also critical that the operating system build not rely strictly on modifications made by the firewall installation process. Firewall installation programs rely on a lowest common denominator approach, so extraneous software packages or modules might not be removed or disabled during the installation process.

In terms of operating system hardening, it is very important that the hardening procedure used during installation has been tailored to the specific operating system undergoing hardening. Some often-overlooked hardening concerns 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 very much 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. Patches and hot fixes are normally released to address security-related issues, so they should be integrated into the firewall build process. One caveat involving patches is that they 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. The practice of configuring multiple network layer (Layer 3) addresses on one physical interface should also be avoided on firewall systems.

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 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 that exists 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 with 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, due to the fact that 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 providing 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 all UNIX and Linux variants.

Once a set of firewall logs has been passed to a centralized logging server, there are quite a few software packages that can be used to examine those logs (several are detailed in the appendix of this document). 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

One of the most nebulous concepts in the information security field involves the answer to a seemingly simple question – What is a security incident?

In general, a security incident is any event in which an unauthorized individual accesses or attempts to access computer systems or resources to which they do not have privileges. It is important to note that there are several different interpretations of these laws, and each of those interpretations is different. These differing interpretations essentially leave it 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. As soon as 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 very seriously. When identified, some organizations or agencies will attempt to prosecute the perpetrator or perpetrators. In all cases, the incidents should be reported¹⁶.

¹⁶ Federal agencies must report security incidents to FedCIRC, the Federal Computer Incident Response Center, at <http://www.fedcirc.gov>.

In essence, the answer to the question 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 incident, 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, and it is extremely important that any such responsibilities are 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 a fairly simple prospect to reconstruct all phases of a security incident.

5.6. Firewall backups

The conduct and maintenance of backups are key points to any firewall administration policy. It is critical that all firewalls are subject to a Day Zero backup – in other words, 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.

As far as the mechanics of firewall backups, it is usually not possible to employ a centralized backup scheme due to the firewall's access control, and 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. This fact means that most firewalls should be built with internal (or external) tape drives. The key is that there should never be tape medium present in the drive unless a backup is being performed.

It is also very desirable (although not always possible) to deploy firewalls that have all critical filesystems burned to CDROM. For UNIX, the only filesystem that actually requires write access is the /var filesystem. All system logs and spool directories can be found in this directory or filesystem. Deployment of Windows NT or Windows 2000 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. Although there is no “perfect” example of these types of firewalls, a good example would be firewalls designed to protect telephone management systems. With the fairly recent rise of in-band PBX management software, firewalls for this function have become very important¹⁷.

Traditionally, PBX resources have been managed using text terminals or proprietary management consoles. Within the last several years, however, it has become very 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 true 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
- Disallow incoming modems
- Disallow outgoing modem
- 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 most organizations is dial-up access. It is not uncommon for users to 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 can be expected to 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.

¹⁷ 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>

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.

A. Links and Resources

(Excerpted from the MIS Training Institute “ISI Swiss Army Knife Reference”,
<http://www.misti.com>)

Books – Internet Security

1. Actually Useful Internet Security Techniques; Larry J. Hughes, Jr.; New Riders Publishing; 1995
2. E-Mail Security; Bruce Schneier; Wiley; 1995
3. Hazards of Hooking Up; Al Berg; LAN Times; June 17, 1996
4. Implementing Internet Security; William Stallings, Peter Stephenson, & Others; New Riders Publishing; 1995
5. Internet Besieged; Dorothy Denning, Peter Denning, et al; Addison-Wesley; 1998
6. Internet Security for Business; Terry Bernstein, Anish Bhimani, Eugene Schultz, Carol Siegel; Wiley; 1996
7. Internet Security: Guide to Web Protection; A Supplement to Infosecurity News; July/August 1996
8. Internet Security - Professional Reference; Numerous Authors; New Riders Publishing; 1996 (includes CD-ROM with security & audit software tools)
9. Internet Security Secrets; John Vacca; IDG Books; 1996
10. Internet Security with Windows NT; Mark Joseph Edwards; Duke Press; 1997
11. Network (In)Security Through IP Packet Filtering; D. Brent Chapman; Proceedings of the Third USENIX UNIX Symposium; September 1992 (also available on NIST BBS)
12. Practical Unix & Internet Security; Simson Garfinkel & Gene Spafford; O'Reilly; 1996
13. Proactive Spam Prevention; Michael Schwager; Sys Admin; March, 1999

Books – Firewall Security

1. Assembly Instructions Included (Cisco Routers); Gilbert Held; Network Magazine; January 2001
2. Building A Floppy Firewall; Andreas Meyer; Sys Admin; January 2001
3. Building Internet Firewalls – 2nd Edition; D. Brent Chapman & Elizabeth D. Zwicky; O'Reilly; 2000
4. Building Linux and OpenBSD Firewalls; Wes Sonnenreich, Tom Yates; Wiley; 2000
5. Cisco IOS: It's Not Just for Routing Anymore; Greg Shipley; Network Computing; May 31, 1999
6. Cisco IOS 12 Network Security; Cisco Press/Macmillan Technical Publishing; 1999

7. Cisco Security Architectures; Gil Held & Kent Hundley; McGraw-Hill; 1999
8. Decipher Your Firewall Logs; Robert Graham; Internet Security Advisor; Mar/Apr 2000
9. Firewall Configuration Done Right; Rik Farrow; Network Magazine; December 1998
10. Firewall Vulnerabilities; Rik Farrow; Network Magazine; August 1999
11. Firewalls 24Seven; Matthew Strebe, Charles Perkins; Sybex Network Press; 1999
12. Firewalls Complete; Marcus Goncalves; McGraw-Hill; 1998 (includes CD-ROM with demo versions of major firewall products)
13. Firewalls & Internet Security - Repelling the Wiley Hacker; Bill Cheswick & Steve Bellovin; Addison-Wesley; 1998
14. FreeBSD Firewall Tools & Techniques; Michael Lucas; Sys Admin; June 2000
15. Great Walls of Fire (Firewall Security); Linda Boyer; NetWare Connection; January 1997
16. The 'Ins' and 'Outs' of Firewall Security; Mike Fratto; Network Computing; September 6, 1999
17. Internet Firewalls & Network Security - Second Edition; Karanjit Siyan; New Riders Publishing; 1996
18. Keeping Your Site Comfortably Secure: An Introduction to Internet Firewalls; NIST Special Publication 800-10
19. Kicking Firewall Tires; Char Sample; Network Magazine; March 1998
20. A Linux Internet Gateway; Marcel Gagne; Sys Admin; June 2000
21. NAT: Network Address Translator; Ron McCarty; Sys Admin; March 2000
22. Packet Filtering and Cisco's Way; Ron McCarty; Sys Admin; May 1999
23. Router-Based Network Defense; Gilbert Held; Sys Admin; March 2000
24. The Use of Routers in Firewall Setup; Matej Sustic; Sys Admin; May 2000

Books – Intrusion Detection & Incident Response

1. Can You Survive A Computer Attack?; Rik Farrow & Richard Power; Network World; May 2000
2. Deploying an Effective Intrusion Detection System; Ramon J. Hontanon; Network Magazine; 2000
3. Detecting Intrusions Within Secured Networks; Dan Sullivan; Internet Security Advisor; Fall 1999
4. FAQ: Network Intrusion Detection Systems; Robert Graham; www.robertgraham.com; March 2000
5. Fcheck: A Solution to Host-Based Intrusion Detection; Ron McCarty; Sys Admin; December 2000
6. An Introduction to Intrusion Detection and Assessment; Rebecca Bace; ICSA; 2000
7. Intrusion Detection: An Introduction to Internet Surveillance, Correlation, Trace Back, Traps & Response; Edward G. Amoroso; Intrusion Net Books; 1998
8. Intrusion Detection; Rebecca Bace; New Riders Publishing; 2000
9. Intrusion Detection: Network Security Beyond the Firewall; Terry Escamilla;

- Wiley; 1998
10. Intrusion Detection Primer; Benjamin J. Thomas; linuxsecurity.com; March 13, 2000
 11. Intrusion Detection Strategies & Design Considerations; Ron McCarty; Sys Admin; September 1999
 12. Investigating Potential Intrusions; Eric Maiwald; Internet Security Advisor; Fall 1999
 13. Snort – A Lock Inside an Intrusion Detection System; Kristy Westphal; Sys Admin; September 2000
 14. Watching the Watchers: Intrusion Detection; Greg Shipley; Network Computing; November 13, 2000

Web Sites – Firewall Security

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

B. Firewall Policy Recommendations

This appendix summarizes the recommendations contained in the main body of this document and adds other general recommendations. This appendix should help technical managers and policy writers create technically sound and maintainable policies that address the major security concerns and problems that firewalls are able to address.

General Recommendations

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

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

A general risk assessment and 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 created, in English, 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 rulebase less error-prone and more verifiable, since the rulebase can be compared to the applications matrix.

Organizations must constantly monitor incident response team reports and security web sites for information about current attacks and vulnerabilities. The firewall policy should be updated as necessary.

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

Recommendations for Firewall Selection

Organizations should invest the appropriate time into researching which firewall and firewall environment is best suited to their needs. There are a number of commercial

sites that deal with firewall selection and analysis; there is a list of evaluated products for use in U.S. Federal Agencies maintained by the National Infrastructure and Protection Center at <http://niap.nist.gov>.

A firewall should be employed that can perform the following general functions:

- Filter packets and protocols
- Perform Stateful inspection of connections
- Perform proxy operations on 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 be operable on the content of SMTP, FTP, and HTTP protocol traffic.

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

Recommendations for Firewall Environment

A border router or other firewall should be used at the Internet connection to create an internal 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 B.1 shows a general picture of a recommended default firewall. 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 so that remote connections can be securely authenticated, as well as encrypted.

Intrusion detection is recommended as an additional safeguard against attacks.

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

Recommendations for Firewall Policy

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

The default policy for the firewall should be to block all traffic 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.

As a general rule, any protocol and traffic that is not necessary, i.e., not used or denied by policy, should be blocked via use of a border 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.

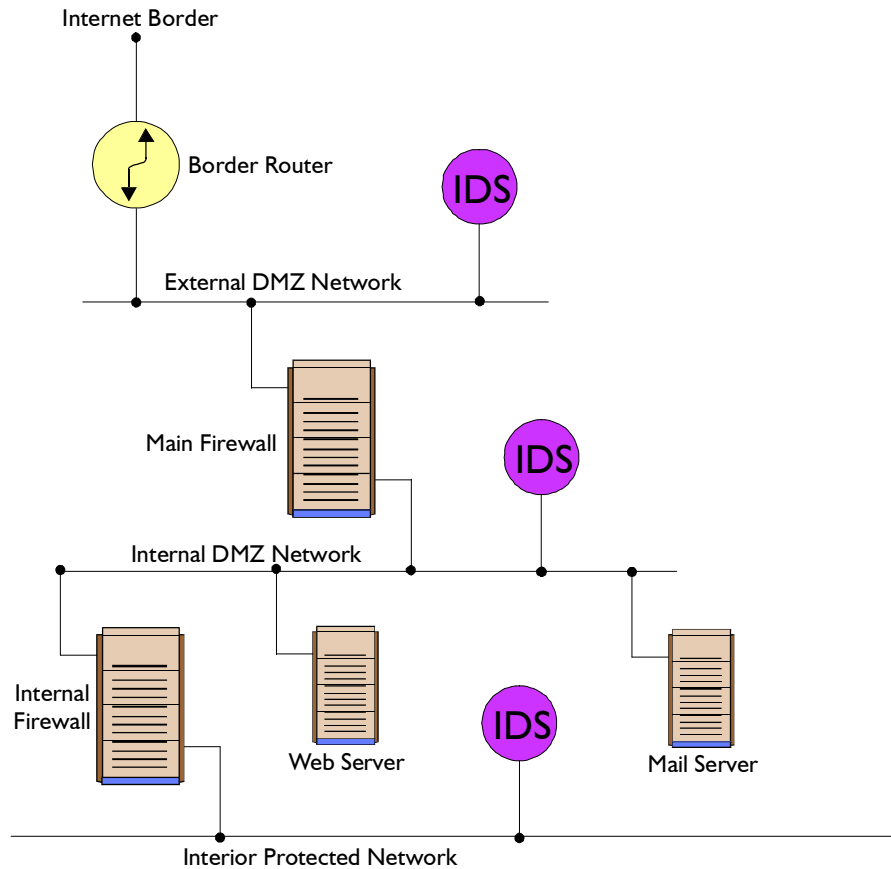


Figure B.1: A Recommended Firewall Environment

Proxy applications should be used to filter HTTP and FTP connections to an organization’s publicly accessible web and FTP servers. Proxy applications should be used for in-bound HTTP connections and for email that are capable of the following operations:

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

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.

Note: the decision to block the above active content, excluding viruses, should be weighed carefully, as blocking active content will render many websites unusable or difficult to use.

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 noted¹⁸:

- Login services
 - telnet (23/tcp) - restricted to specific systems using strong authentication
 - SSH (22/tcp) - restricted to specific systems
 - FTP (21/tcp) - restricted to specific systems using strong authentication
 - NetBIOS (139/tcp) - always block
 - rlogin et al (512/tcp through 514/tcp) - always block
- RPC and NFS
 - Portmap/rpcbind (111/tcp and 111/udp) - always block
 - NFS (2049/tcp and 2049/udp) - always block
 - lockd (4045/tcp and 4045/udp) - always block
- NetBIOS in Windows NT
 - 135 (tcp and udp) - always block
 - 137 (udp) - always block
 - 138 (udp) - always block
 - 139 (tcp) - always block
 - 445 (tcp and udp) for Windows 2000 - always block
- .X Windows
 - 6000/tcp through 6255/tcp - always block
- Naming services
 - DNS (53/udp) - restrict to DNS servers
 - DNS zone transfers (53/tcp) - block unless from external secondaries
 - LDAP (389/tcp and 389/udp) - always block
- Mail
 - SMTP (25/tcp) - block unless from 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.)

¹⁸ 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>.

- “Small Services”
 - ports below 20/tcp and 20/udp - always block
 - time (37/tcp and 37/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 and 161/udp, 162/tcp and 162/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.

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 containing ICMP (Internet Control Message Protocol) traffic.
- 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 traffic containing directed broadcast addresses.

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 applied¹⁹.

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 incidents²⁰.

¹⁹ 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.

²⁰ The Federal Computer Incident Response Center (FedCIRC) is the central coordination and analysis facility dealing with computer security related issues affecting the civilian agencies and departments of the Federal Government. See <http://fedcirc.gov>.