```
firewalls is generally at the application level, rather than at a
network level.

Router-based Firewall:
A firewall where the security is implemented using screening
routers as the primary means of protecting the network.

Screening Router:
A router that is used to implement part of the security of a
firewall by configuring it to selectively permit or deny
traffic at a network level.

Bastion Host:
A host system that is a "strong point" in the network's security
perimeter. Bastion hosts should be configured to be particularly
resistant to attack. In a host-based firewall, the bastion host
is the platform on which the firewall software is run.
Bastion hosts are also referred to as "gateway hosts."

Dual-Homed Gateway:
A firewall consisting of a bastion host with 2 network interfaces,
one of which is connected to the protected network, the other of
which is connected to the Internet. IP traffic forwarding is
usually disabled, restricting all traffic between the two networks
to whatever passes through some kind of application proxy.

Application Proxy:
An application that forwards application traffic through a
firewall. Proxies tend to be specific to the protocol they
are designed to forward, and may provide increased access
control or audit.

Screened Subnet:
A firewall architecture in which a "sand box" or "demilitarized
zone" network is set up between the protected network and the
Internet, with traffic between the protected network and the
Internet blocked. Conceptually, this is similar to a dual-homed
gateway, except that an entire network, rather than a single
host is reachable from the outside.




Contributors:
-------------
mjr@tis.com - Marcus Ranum, Trusted Information Systems
leibowa@wl.com - Allen Leibowitz, Warner Lambert Inc.
brent@greatcircle.com - Brent Chapman, Great Circle Associates
bdboyle@erenj.com - Brian Boyle, Exxon Research
```

fingering.

Many sites block inbound finger requests for a variety of reasons, foremost being past security bugs in the finger server (the Morris internet worm made these bugs famous) and the risk of proprietary or sensitive information being revealed in user's finger information.

Date: Thu Mar 3 12:40:54 1994
From: Fwalls-FAQ@tis.com
Subject: 16: How do I make gopher, archie, and other services work through my firewall?

This is still an area of active research in the firewall community. Many firewall administrators support these services only through the character-cell interface provided by telnet. Unfortunately, many of the sexier network services make connections to multiple remote systems, without transmitting any inline information that a proxy could take advantage of, and often the newer information retrieval systems transmit data to local hosts and disks with only minimal security. There are risks that (for example) WAIS clients may request uuencoded files, which decode and modify security related files in the user's home directory. At present, there is a lot of head-scratching going on between the firewall administrators who are responsible for guarding the network perimeters, and the users, who want to take advantage of these very sexy and admittedly useful tools.

Date: Mon Jun 6 10:12:03 1994
From: Fwalls-FAQ@tis.com
Subject: 17: What are the issues about X-Window through a firewall?

X Windows is a very useful system, but unfortunately has some major security flaws. Remote systems that can gain or spoof access to a workstation's X display can monitor keystrokes that a user enters, download copies of the contents of their windows, etc.

While attempts have been made to overcome them (E.g., MIT "Magic Cookie") it is still entirely too easy for an attacker to interfere with a user's X display. Most firewalls block all X traffic. Some permit X traffic through application proxies such as the DEC CRL X proxy (FTP crl.dec.com).

Date: Thu Mar 24 14:05:27 1994
From: Fwalls-FAQ@tis.com
Subject: 18: Glossary of firewall related terms

Host-based Firewall:
A firewall where the security is implemented in software running on a general-purpose computer of some sort. Security in host-based

```
like ftp.uu.net that insist on having a name for the
machines they talk to. This may fail when talking to sites
that do a DNS cross-check in which the host name is matched
against its address and vice versa.

Note that hiding names in the DNS doesn't address the
problem of host names "leaking" out in mail headers,
news articles, etc.

------------------------------

Date: Thu Mar 3 21:14:24 1994
From: Fwalls-FAQ@tis.com
Subject: 13: How do I make FTP work through my firewall?

Generally, making FTP work through the firewall is done
either using a proxy server or by permitting incoming
connections to the network at a restricted port range, and
otherwise restricting incoming connections using something
like "established" screening rules. The FTP client is then
modified to bind the data port to a port within that range.
This entails being able to modify the FTP client application
on internal hosts.

A different approach is to use the FTP "PASV"
option to indicate that the remote FTP server should permit
the client to initiate connections. The  PASV approach
assumes that the FTP server on the remote system supports
that operation. (See RFC1579 for more information)

Other sites prefer to build client versions of
the FTP program that are linked against a SOCKS library.

------------------------------

Date: Mon Mar 7 13:00:08 1994
From: Fwalls-FAQ@tis.com
Subject: 14: How do I make Telnet work through my firewall?

Telnet is generally supported either by using an application
proxy, or by simply configuring a router to permit outgoing
connections using something like the "established" screening
rules. Application proxies could be in the form of a standalone
proxy running on the bastion host, or in the form of a SOCKS
server and a modified client.

------------------------------

Date: Thu Mar 3 14:16:12 1994
From: Fwalls-FAQ@tis.com
Subject: 15: How do I make Finger and whois work through my firewall?

Permit connections to the finger port from only trusted
machines, which can issue finger requests in the form of:
finger user@host.domain@firewall

This approach only works with the standard UNIX version of
finger. Some finger servers do not permit user@host@host
```

```
From: Fwalls-FAQ@tis.com
Subject: 12: How do I make DNS work with a firewall?

Some organizations want to hide DNS names from the outside.
Many experts disagree as to whether or not hiding DNS names
is worthwhile, but if site/corporate policy mandates hiding
domain names, this is one approach that is known to work.

This approach is one of many, and is useful for
organizations that wish to hide their host names from the
Internet. The success of this approach lies on the fact that
DNS clients on a machine don't have to talk to a DNS server
on that same machine.  In other words, just because there's
a DNS server on a machine, there's nothing wrong with (and
there are often advantages to) redirecting that machine's
DNS client activity to a DNS server on another machine.

First, you set up a DNS server on the bastion host that the
outside world can talk to. You set this server up so that it
claims to be authoritative for your domains.  In fact, all
this server knows is what you want the outside world to
know; the names and addresses of your gateways, your
wildcard MX records, and so forth.  This is the "public"
server.

Then, you set up a DNS server on an internal machine.  This
server also claims to be authoritiative for your domains;
unlike the public server, this one is telling the truth.
This is your "normal" nameserver, into which you put all
your "normal" DNS stuff.  You also set this server up to
forward queries that it can't resolve to the public server
(using a "forwarders" line in /etc/named.boot on a UNIX
machine, for example).

Finally, you set up all your DNS clients (the
/etc/resolv.conf file on a UNIX box, for instance),
including the ones on the machine with the public server, to
use the internal server.  This is the key.

An internal client asking about an internal host asks the
internal server, and gets an answer; an internal client
asking about an external host asks the internal server,
which asks the public server, which asks the Internet, and
the answer is relayed back.  A client on the public server
works just the same way.  An external client, however,
asking about an internal host gets back the "restricted"
answer from the public server.

This approach assumes that there's a packet filtering
firewall between these two servers that will allow them to
talk DNS to each other, but otherwise restricts DNS between
other hosts.

Another trick that's useful in this scheme is to employ
wildcard PTR records in your IN-ADDR.ARPA domains. These
cause an an address-to-name lookup for any of your non-
public hosts to return something like "unknown.YOUR.DOMAIN"
rather than an error.  This satisfies anonymous FTP sites
```

```
     9  ip access-group 11
    10  !
    11  access-list 10 permit ip 128.88.254.2 0.0.0.0
         128.88.0.0 0.0.255.255
    12  access-list 10 deny   tcp 0.0.0.0 255.255.255.255
         128.88.0.0 0.0.255.255 lt 1025
    13  access-list 10 deny   tcp 0.0.0.0 255.255.255.255
         128.88.0.0 0.0.255.255 gt 4999
    14  access-list 10 permit tcp 0.0.0.0 255.255.255.255
         128.88.0.0 0.0.255.255
    15  !
    16  access-list 11 permit ip 128.88.0.0 0.0.255.255
         128.88.254.2 0.0.0.0
    17  access-list 11 deny   tcp 128.88.0.0 0.0.255.255
         0.0.0.0 255.255.255.255 eq 25
    18  access-list 11 permit tcp 128.88.0.0 0.0.255.255
         0.0.0.0 255.255.255.255
```

```
Lines   Explanation
=====   ===========
    1   Although this is not a filtering rule, it is good to include here.

    5   Ethernet 0 is on the red net.  Extended access list 10 will
        be applied to output on this interface.  You can also
        think of output from the red net as input on the blue net.

    9   Ethernet 1 is on the blue net.  Extended access list 11 will
        be applied to output on this interface.

   11   Allow all traffic from the gateway machine to the blue net.

12-14   Allow connections originating from the red net that come in
        between ports 1024 and 5000.  This is to allow ftp data
        connections back into the blue net.  5000 was chosen as the
        upper limit as it is where OpenView starts.

        Note: again, we are assuming this is acceptable for the given policy.
              There is no way to tell a Cisco to filter on source port.
              Newer versions of the Cisco firmware will apparently support
              source port filtering.

        Since the rules are tested until the first match we must use this
        rather obtuse syntax.

   16   Allow all blue net packets to the gateway machine.

   17   Deny SMTP (tcp port 25) mail to the red net.

   18   Allow all other TCP traffic to the red net.
```

Cisco.Com has an archive of examples for building firewalls
using Cisco routers, available for FTP from: ftp.cisco.com
in /pub/acl-examples.tar.Z

Date: Thu Mar 3 13:52:47 1994

shows the implementation of a specific policy. Your policy will undoubtedly vary.

In this example, a company has Class B network address of 128.88.0.0 and is using 8 bits for subnets. The Internet connection is on the "red" subnet 128.88.254.0. All other subnets are considered trusted or "blue" subnets.

```
     +---------------+ +---------------+
     | IP provider   | |   Gateway     |
     | 128.88.254.1  | | 128.88.254.2  |
     +------+--------+ +------+--------+
            |                              "Red" net
  ----------+-----------------+----------------------------------
                              |
                       +------+--------+
                       |   Cisco       |
                       | 128.88.254.3  |
                       |...............|
                       | 128.88.1.1    |
                       +---------------+
                              |
  ----------------------------+----------------------------------
            |                              "Blue" net
     +------+--------+
     | mail router   |
     | 128.88.1.2    |
     +---------------+
```

Keeping the following points in mind will help in understanding the configuration fragments:

1. Ciscos applying filtering to output packets only.
2. Rules are tested in order and stop when the first match is found.
3. There is an implicit deny rule at the end of an access list that denies everything.

The example below concentrates on the filtering parts of a configuration. Line numbers and formatting have been added for readability.

The policy to be implemented is:
- Anything not explicitly allowed is denied
- Traffic between the external gateway machine and blue net hosts is allowed.
- permit services orginating from the blue net
- allow a range of ports for FTP data connections back to the blue net.

```
no ip source-route
!
interface Ethernet 0
ip address 128.88.1.1 255.255.255.0
ip access-group 10
!
interface Ethernet 1
ip address 128.88.254.3 255.255.255.0
```

```
level of service that may be provided (since a proxy needs
to be developed for each desired service). The old trade-off
between ease-of-use and security comes back to haunt us with
a vengeance.

------------------------------

Date: Thu Mar 10 16:56:35 1994
From: Fwalls-FAQ@tis.com
Subject: 9: What are proxy servers and how do they work?

A proxy server (sometimes referred to as an application
gateway or forwarder) is an application that mediates
traffic between a protected network and the Internet.
Proxies are often used instead of router-based traffic
controls, to prevent traffic from passing directly between
networks. Many proxies contain extra logging or support for
user authentication. Since proxies must "understand" the
application protocol being used, they can also implement
protocol specific security (e.g., an FTP proxy might be
configurable to permit incoming FTP and block outgoing
FTP).

Proxy servers are application specific. In order to support
a new protocol via a proxy, a proxy must be developed for
it. SOCKS is a generic proxy system that can be compiled
into a client-side application to make it work through a
firewall. Its advantage is that it's easy to use, but it
doesn't support the addition of authentication hooks or
protocol specific logging. For more information on SOCKS,
see ftp.nec.com: /pub/security/socks.cstc   Users are
encouraged to check the file "FILES" for a description
of the directory's contents.


------------------------------

Date: Mon Jun 6 10:07:36 1994
From: Fwalls-FAQ@tis.com
Subject: 10: What are some cheap packet screening tools?

The Texas AMU security tools include software for
implementing screening routers (FTP net.tamu.edu,
pub/security/TAMU).  Karlbridge is a PC-based screening
router kit (FTP nisca.acs.ohio-state.edu, pub/kbridge). A
version of the Digital Equipment Corporation "screend"
kernel screening software is available for BSD/386,
NetBSD, and BSDI. Many commercial routers support screening
of various forms.

------------------------------

Date: Mon Jun 6 10:05:51 1994
From: Fwalls-FAQ@tis.com
Subject: 11: What are some reasonable filtering rules for my Cisco?

The following example shows one possible configuration for
using the Cisco as a filtering router.  It is a sample that
```

addressed by the lucky person who has been tasked with the responsibility of designing, specifying, and implementing or overseeing the installation of a firewall.

The first and most important is reflects the policy of how your company or organization wants to operate the system: is the firewall in place to explicitly deny all services except those critical to the mission of connecting to the net, or is the firewall in place to provide a metered and audited method of "queuing" access in a non-threatening manner. There are degrees of paranoia between these positions; the final stance of your firewall may be more the result of a political than an engineering decision.

The second is: what level of monitoring, redundancy, and control do you want? Having established the acceptable risk level (e.g.: how paranoid you are) by resolving the first issue, you can form a checklist of what should be monitored, permitted, and denied. In other words, you start by figuring out your overall objectives, and then combine a needs analysis with a risk assessment, and sort the almost always conflicting requirements out into a laundry list that specifies what you plan to implement.

The third issue is financial. We can't address this one here in anything but vague terms, but it's important to try to quantify any proposed solutions in terms of how much it will cost either to buy or to implement. For example, a complete firewall product may cost between $100,000 at the high end, and free at the low end. The free option, of doing some fancy configuring on a Cisco or similar router will cost nothing but staff time and cups of coffee. Implementing a high end firewall from scratch might cost several man-months, which may equate to $30,000 worth of staff salary and benefits. The systems management overhead is also a consideration. Building a home-brew is fine, but it's important to build it so that it doesn't require constant and expensive fiddling-with. It's important, in other words, to evaluate firewalls not only in terms of what they cost now, but continuing costs such as support.

On the technical side, there are a couple of decisions to make, based on the fact that for all practical purposes what we are talking about is a static traffic routing service placed between the network service provider's router and your internal network. The traffic routing service may be implemented at an IP level via something like screening rules in a router, or at an application level via proxy gateways and services.

The decision to make here is whether to place an exposed stripped-down machine on the outside network to run proxy services for telnet, ftp, news, etc., or whether to set up a screening router as a filter, permitting communication with one or more internal machines. There are plusses and minuses to both approaches, with the proxy machine providing a greater level of audit and potentially security in return for increased cost in configuration and a decrease in the

```
There are several books that touch on firewalls. The best
known are:

Cheswick and Bellovin, "Firewalls and Internet Security:
Repelling the Wily Hacker"  Addison-Wesley, ??, 1994

Garfinkel  and Spafford, "Practical UNIX Security"  O'Reilly
and associates (discusses primarily host security)

Related references are:

Comer and Stevens, "Internetworking with TCP/IP" Prentice Hall, 1991

Curry, "UNIX System Security" Addison Wesley, 1992

------------------------------

Date: Thu Mar 3 13:48:14 1994
From: Fwalls-FAQ@tis.com
Subject: 6: Where can I get more information on firewalls on the network?

Ftp.greatcircle.com - Firewalls mailing list archives.
Directory: pub/firewalls

Ftp.tis.com - Internet firewall toolkit and papers.
Directory: pub/firewalls

Research.att.com - Papers on firewalls and breakins.
Directory: dist/internet_security

Net.Tamu.edu - Texas AMU security tools.
Directory: pub/security/TAMU

The internet firewalls mailing list is a forum for firewall
administrators and implementors. To subscribe to Firewalls, send
"subscribe firewalls"
in the body of a message (not on the "Subject:" line) to
"Majordomo@GreatCircle.COM". Archives of past Firewalls postings are
available for anonymous FTP from ftp.greatcircle.com in pub/firewalls/archive

------------------------------

Date: Thu Mar 3 12:38:10 1994
From: Fwalls-FAQ@tis.com
Subject: 7: What are some commercial products or consultants who sell/service firewalls?

We feel this topic is too sensitive to address in a FAQ, as
well as being difficult to maintain an up-to-date list.


------------------------------

Date: Thu Mar 3 12:38:31 1994
From: Fwalls-FAQ@tis.com
Subject: 8: What are some of the basic design decisions in a firewall?

There are a number of basic design issues that should be
```

Subject: 3: What can a firewall protect against?

Some firewalls permit only Email traffic through them, thereby protecting the network against any attacks other than attacks against the Email service. Other firewalls provide less strict protections, and block services that are known to be problems.

Generally, firewalls are configured to protect against unauthenticated interactive logins from the "outside" world. This, more than anything, helps prevent vandals from logging into machines on your network. More elaborate firewalls block traffic from the outside to the inside, but permit users on the inside to communicate freely with the outside. The firewall can protect you against any type of network borne attack if you unplug it.

Firewalls are also important since they can provide a single "choke point" where security and audit can be imposed. Unlike in a situation where a computer system is being attacked by someone dialing in with a modem, the firewall can act as an effective "phone tap" and tracing tool.

Date: Thu Mar 3 14:02:07 1994
From: Fwalls-FAQ@tis.com
Subject: 4: What can't a firewall protect against?

Firewalls can't protect against attacks that don't go through the firewall. Many corporations that connect to the Internet are very concerned about proprietary data leaking out of the company through that route. Unfortunately for those concerned, a magnetic tape can just as effectively be used to export data. Firewall policies must be realistic, and reflect the level of security in the entire network. For example, a site with top secret or classified data doesn't need a firewall at all: they shouldn't be hooking up to the internet in the first place, or the systems with the really secret data should be isolated from the rest of the corporate network.

Firewalls can't protect very well against things like viruses. There are too many ways of encoding binary files for transfer over networks, and too many different architectures and viruses to try to search for them all. In other words, a firewall cannot replace security-consciousness on the part of your users. In general, a firewall cannot protect against a data-driven attack -- attacks in which something is mailed or copied to an internal host where it is then executed. This form of attack has occurred in the past against various versions of Sendmail.

Date: Thu Mar 24 13:46:32 1994
From: Fwalls-FAQ@tis.com
Subject: 5: What are good sources of print information on firewalls?

```
16: How do I make gopher, archie, and other services work through my firewall?
17: What are the issues about X-Window through a firewall?
18: Glossary of firewall related terms

------------------------------

Date: Thu Mar 3 12:35:59 1994
From: Fwalls-FAQ@tis.com
Subject: 1: What is a network firewall?

A firewall is any one of several ways of protecting one
network from another untrusted network. The actual mechanism
whereby this is accomplished varies widely, but in
principle, the firewall can be thought of as a pair of
mechanisms: one which exists to block traffic, and the other
which exists to permit traffic. Some firewalls place a
greater emphasis on blocking traffic, while others emphasize
permitting traffic.

------------------------------

Date: Thu Mar 3 12:36:15 1994
From: Fwalls-FAQ@tis.com
Subject: 2: Why would I want a firewall?

The Internet, like any other society, is plagued with the
kind of jerks who enjoy the electronic equivalent of writing
on other people's walls with spraypaint, tearing their
mailboxes off, or just sitting in the street blowing their
car horns. Some people try to get real work done over the
Internet, and others have sensitive or proprietary data they
must protect. A firewall's purpose is to keep the jerks out
of your network while still letting you get your job done.

Many traditional-style corporations and data centers have
computing security policies and practices that must be
adhered to. In a case where a company's policies dictate how
data must be protected, a firewall is very important, since
it is the embodiment of the corporate policy. Frequently,
the hardest part of hooking to the Internet, if you're a
large company, is not justifying the expense or effort, but
convincing management that it's safe to do so. A firewall
provides not only real security - it often plays an
important role as a security blanket for management.

Lastly, a firewall can act as your corporate "ambassador" to
the Internet. Many corporations use their firewall systems
as a place to store public information about corporate
products and services, files to download, bug-fixes, and so
forth. Several of these systems have become important parts
of the Internet service structure (e.g.: UUnet.uu.net,
gatekeeper.dec.com) and have reflected well on their
corporate sponsors.

------------------------------

Date: Thu Mar 3 13:24:13 1994
From: Fwalls-FAQ@tis.com
```

B

Internet Firewalls Frequently Asked Questions

This appendix contains a FAQ (Frequently Asked Questions) on Internet firewalls. It is available on-line at several locations, including

ftp.greatcircle.com

ftp.tis.com

Internet Firewalls Frequently Asked Questions

```
About the FAQ
=============
This FAQ is not an advertisement or endorsement for any
product, company, or consultant. The maintainer welcomes input
and comments on the contents of this FAQ. Comments related
to the FAQ should be addressed to Fwalls-FAQ@tis.com.


Contents:
=========
1: What is a network firewall?
2: Why would I want a firewall?
3: What can a firewall protect against?
4: What can't a firewall protect against?
5: What are good sources of print information on firewalls?
6: Where can I get more information on firewalls on the  network?
7: What are some commercial products or consultants who sell/service firewalls?
8: What are some of the basic design decisions in a firewall?
9: What are proxy servers and how do they work?
10: What are some cheap packet screening tools?
11: What are some reasonable filtering rules for my Cisco?
12: How do I make DNS work with a firewall?
13: How do I make FTP work through my firewall?
14: How do I make Telnet work through my firewall?
15: How do I make Finger and whois work through my firewall?
```

To access the clearinghouse via an http client, use the following Uniform Resource Locator (URL):

http://www.first.org/first

More information about FIRST can be obtained from any participating member or the National Institute of Standards and Technology at the following address:

National Institute of Standards and Technology
Forum of Incident Response and Security Teams
A-216, Technology
Gaithersburg, MD 20899
301-975-3359
first@first.org

privacy, computer viruses, advanced authentication, and firewalls. The clearinghouse can be accessed via the Internet (http, gopher, and ftp) and dial-in. To connect via gopher and ftp, use the following:

gopher csrc.ncsl.nist.gov

ftp csrc.ncsl.nist.gov – login as user "anonymous"

To access the clearinghouse via an http client, use the following Uniform Resource Locator (URL):

http://csrc.ncsl.nist.gov

The clearinghouse can be accessed via modem, at V.34 (28.8), V.32 (14.4), and lower speeds. The clearinghouse assumes 8 bit characters, no parity, and 1 stop bit. Dial the following:

301-948-5717

and you will be connected to a lynx http client. This client is self-explanatory to use; on-line help is included. A full assortment of download protocols are available for transferring files to your local system.

A.3 Forum of Incident Response and Security Teams

The Forum of Incident Response and Security Teams (FIRST) is an organization whose members work together voluntarily to deal with computer security problems and their prevention. The organization is composed of incident response teams, a steering committee, and a secretariat, which is currently NIST, and ad hoc working groups. Much of the focus is on Internet security-related threats. The forum meets regularly and conducts annual workshops on incident handling.

Many businesses, universities, and government organizations are members of FIRST. A list of members, background information, and information on membership is available on-line. To connect via gopher and ftp, use the following:

gopher gopher.first.org

ftp first.org – login as user "anonymous"

A

On-Line Sources for More Information

Readers who need additional information on firewalls, Internet security, and security policy issues should consult the references. In addition, there are several on-line sources for more information.

A.1 Firewall-Specific Information

Readers are urged to consult the following ftp site,

ftp.greatcircle.com

for more information on firewalls. This site contains information on firewall vendors, firewall-related papers and articles, and collections of mailing list postings organized by topic. This site also maintains a mailing list on firewall issues; information about the mailing list is available on-line at the site.

In addition to the above, a number of router and firewall vendors maintain e-mail lists and ftp sites that contain firewall and Internet security-related information. Check with your vendor for the appropriate address.

A.2 NIST Computer Security Resource Clearinghouse

NIST operates a clearinghouse of computer security-related information. This clearinghouse contains information on a broad range of subjects, including computer security incident response team alerts, papers on Internet security, policy and training information,

[Cur92] David Curry. *UNIX System Security: A Guide for Users and System Administrators*. Addison-Wesley, Reading, MA, 1992.

[Farm93] Dan Farmer and Wietse Venema. Improving the security of your site by breaking into it. Available from FTP.WIN.TUE.NL, file /pub/security/admin-guide-to-cracking.101.Z, 1993.

[Ford94] Warwick Ford. *Computer Communications Security*. Prentice-Hall, Englewood Cliffs, NJ, 1994.

[Garf92] Simpson Garfinkel and Gene Spafford. *Practical UNIX Security*. O'Reilly and Associates, Inc., Sebastopol, CA, 1992.

[Haf91] Katie Hafner and John Markoff. *Cyberpunk: Outlaws and Hackers on the Computer Frontier*. Simon and Schuster, New York, 1991.

[Hunt92] Craig Hunt. *TCP/IP Network Administration*. O'Reilly and Associates, Inc., Sebastopol, CA, 1992.

[NIST91a] NIST. Advanced Authentication Technology. CSL Bulletin, National Institute of Standards and Technology, November 1991.

[NIST91b] NIST. Establishing a Computer Security Incident Response Capability. Special Publication 800-3, National Institute of Standards and Technology, January 1991.

[NIST93] NIST. Connecting to the Internet: Security Considerations. CSL Bulletin, National Institute of Standards and Technology, July 1993.

[NIST94a] NIST. Guideline for the use of Advanced Authentication Technology Alternatives. Federal Information Processing Standard 190, National Institute of Standards and Technology, September 1994.

[NIST94b] NIST. Reducing the Risk of Internet Connection and Use. CSL Bulletin, National Institute of Standards and Technology, May 1994.

[NIST94c] NIST. Security in Open Systems. Special Publication 800-7, National Institute of Standards and Technology, September 1994.

[Ran93] Marcus Ranum. Thinking About Firewalls. In *SANS-II Conference*, April 1993.

[RFC1244] Paul Holbrook and Joyce Reynolds. RFC 1244: Security Policy Handbook. prepared for the Internet Engineering Task Force, 1991.

Bibliography

[Avol94] Frederick Avolio and Marcus Ranum. A Network Perimeter With Secure Internet Access. In *Internet Society Symposium on Network and Distributed System Security*, pages 109–119. Internet Society, February 2-4 1994.

[Bel89] Steven M. Bellovin. Security Problems in the TCP/IP Protocol Suite. *Computer Communications Review*, 9(2):32–48, April 1989.

[Cerf93] Vinton Cerf. A National Information Infrastructure. *Connexions*, June 1993.

[CERT94] Computer Emergency Response Team/Coordination Center. CA-94:01, Ongoing Network Monitoring Attacks. Available from FIRST.ORG, file pub/alerts/cert9401.txt, February 1994.

[Chap92] D. Brent Chapman. Network (In)Security Through IP Packet Filtering. In *USENIX Security Symposium III Proceedings*, pages 63–76. USENIX Association, September 14-16 1992.

[Ches94] William R. Cheswick and Steven M. Bellovin. *Firewalls and Internet Security.* Addison-Wesley, Reading, MA, 1994.

[CIAC94a] Computer Incident Advisory Capability. Number e-07, unix sendmail vulnerabilities update. Available from FIRST.ORG, file pub/alerts/e-07.txt, January 1994.

[CIAC94b] Computer Incident Advisory Capability. Number e-09, network monitoring attacks. Available from FIRST.ORG, file pub/alerts/e-09.txt, February 1994.

[CIAC94c] Computer Incident Advisory Capability. Number e-14, wuarchive ftpd trojan horse. Available from FIRST.ORG, file pub/alerts/e-14.txt, February 1994.

[Com91a] Douglas E. Comer. *Internetworking with TCP/IP: Principles, Protocols, and Architecture.* Prentice-Hall, Englewood Cliffs, NJ, 1991.

[Com91b] Douglas E. Comer and David L. Stevens. *Internetworking with TCP/IP: Design, Implementation, and Internals.* Prentice-Hall, Englewood Cliffs, NJ, 1991.

- standardize operating system versions and software to make installation of patches and security fixes more manageable,
- institute a program for efficient, site-wide installation of patches and new software,
- use services to assist in centralizing system administration, if this will result in better administration and better security,
- perform periodic scans and checks of host systems to detect common vulnerabilities and errors in configuration, and
- ensure that a communications pathway exists between system administrators and firewall/site security administrators to alert the site about new security problems, alerts, patches, and other security-related information.

4.3.3 Incident Handling Contacts

An important consideration under firewall and site system administration is incident handling assistance and contacts. NIST recommends that organizations develop incident handling capabilities that can deal with suspicious activity and intrusions, and that can keep an organization up to date with computer security threat and vulnerability information. Because of the changing nature of Internet threats and risks, it is important that those maintaining firewalls be part of the incident handling process. Firewall administrators need to be aware of new vulnerabilities in products they are using, or if intruder activity is on-going and can be detected using prescribed techniques. [Cur92], [Garf92], and [RFC1244], contain information on developing incident response teams and contacts. NIST has produced a publication specifically on creating incident response capabilities [NIST91b].

See Appendix A for more information on incident response team contacts and the Forum of Incident Response and Security Teams (FIRST).

Many vendors offer maintenance services along with firewall installation, therefore the organization should consider whether it has the internal resources to perform the above.

4.3 Administration Issues with Firewalls

It should not be surprising that firewall administration is a critical job role and should be afforded as much time as possible. In small organizations, it may require less than a full-time position, however it should take precedence over other duties. The cost of a firewall should include the cost of administering the firewall; administration should never be shortchanged.

4.3.1 System Management Expertise

As evidenced by previous discussions concerning the many host system break-ins occurring throughout the Internet, the need for highly trained, quality, full-time host system administrators is clearly shown. But, there is also indication that this need is not being met; many sites do not manage systems such that the systems are secure and protected from intruder attacks. Many system managers are part-time at best and do not upgrade systems with patches and bug fixes as available.

Firewall management expertise is a highly critical job role, as a firewall can only be as effective as its administration. If the firewall is not maintained properly, it may become insecure, and may permit break-ins while providing an illusion that the site is still secure. A site's security policy should clearly reflect the importance of strong firewall administration. Management should demonstrate its commitment to this importance in terms of full-time personnel, proper funding for procurement and maintenance and other necessary resources.

4.3.2 Site System Administration

A firewall is not an excuse to pay less attention to site system administration. It is in fact the opposite: if a firewall is penetrated, a poorly administered site could be wide-open to intrusions and resultant damage. A firewall in no way reduces the need for highly skilled system administration.

At the same time, a firewall can permit a site to be "proactive" in its system administration as opposed to reactive. Because the firewall provides a barrier, sites can spend more time on system administration duties and less time reacting to incidents and damage control. It is recommended that sites

any firewall installation. Therefore, flexibility to adapt to changing needs is an important consideration.

4.2.2 To Buy or Build a Firewall

A number of organizations may have the capability to build a firewall for themselves, i.e., put together a firewall by using available software components and equipment or by writing a firewall from scratch. At the same time, there are a number of vendors offering a wide spectrum of services in firewall technology. Service can be as limited as providing the necessary hardware and software only, or as broad as providing services to develop security policy, risk assessments, security reviews and security training.

Whether one buys or builds a firewall it must be reiterated that one should first develop a policy and related requirements before proceeding. If an organization is having difficulty developing a policy, it may need to contact a vendor who can assist in this process.

If an organization has the in-house expertise to build a firewall, it may prove more cost-effective to do so. One of the advantages of building a firewall is that in-house personnel understand the specifics of the design and use of the firewall. This knowledge may not exist in-house with a vendor supported firewall.

At the same time, an in-house firewall can be expensive in terms of time required to build and document the firewall, and the time required for maintaining the firewall and adding features to it as required. These costs are sometimes not considered; organizations sometimes make the mistake of counting only the costs for the equipment. If a true accounting is made for all costs associated with building a firewall, it could prove more economical to purchase a vendor firewall.

In deciding whether to purchase or build a firewall, answers to the following questions may help an organization gauge whether it has the resources to build and operate a successful firewall:

- how will the firewall be tested; who will verify that the firewall performs as expected,
- who will perform general maintenance of the firewall, such as backups and repairs,
- who will install updates to the firewall, such as for new proxy servers, new patches, and other enhancements,
- can security-related patches and problems be corrected in a timely manner, and
- who will perform user support and training.

- The firewall should employ filtering techniques to permit or deny services to specified host systems as needed.
- The IP filtering language should be flexible, user-friendly to program, and should filter on as many attributes as possible, including source and destination IP address, protocol type, source and destination TCP/UDP port, and inbound and outbound interface.
- The firewall should use proxy services for services such as FTP and TELNET, so that advanced authentication measures can be employed and centralized at the firewall. If services such as NNTP, X, http, or gopher are required, the firewall should contain the corresponding proxy services.
- The firewall should contain the ability to centralize SMTP access, to reduce direct SMTP connections between site and remote systems. This results in centralized handling of site e-mail.
- The firewall should accomodate public access to the site, such that public information servers can be protected by the firewall but can be segregated from site systems that do not require the public access.
- The firewall should contain the ability to concentrate and filter dial-in access.
- The firewall should contain mechanisms for logging traffic and suspicious activity, and should contain mechanisms for log reduction so that logs are readable and understandable.
- If the firewall requires an operating system such as UNIX, a secured version of the operating system should be part of the firewall, with other security tools as necessary to ensure firewall host integrity. The operating system should have all patches installed.
- The firewall should be developed in a manner that its strength and correctness is verifiable. It should be simple in design so that it can be understood and maintained.
- The firewall and any corresponding operating system should be updated with patches and other bug fixes in a timely manner.

There are undoubtably more issues and requirements, however many of them will be specific to each site's own needs. A thorough requirements definition and high-level risk assessment will identify most issues and requirements, however it should be emphasized that the Internet is a constantly changing network. New vulnerabilities can arise, and new services and enhancements to other services may represent potential difficulties for

One can make a useful distinction that information server traffic, i.e., the traffic concerned with retrieving information from an organization's information server, is fundamentally different from other "conduct of business" traffic such as e-mail (or other information server traffic for the purposes of business research). The two types of traffic have their own risks and do not necessarily need to be mixed with each other.

Section 3 discusses incorporating an information server into the firewall design. The screened subnet and dual-homed gateway firewall examples show information servers that can be located on a screened subnet and in effect be isolated from other site systems. This reduces the chance that an information server could be compromised and then used to attack site systems.

4.2 Procuring a Firewall

After policy has been decided, there are a number of issues to be considered in procuring a firewall. Many of these issues are the same as for procuring other software systems, thus familiar steps such as requirements definition, analysis, and design specification are standard. The following sections describe some additional considerations, including minimal criteria for a firewall and whether to build or purchase a firewall.

4.2.1 What Should a Firewall Contain?

Once the decision is made to use firewall technology to implement an organization's security policy, the next step is to procure a firewall that provides the appropriate level of protection and is cost-effective. However, what features should a firewall have, at a minimum, to provide effective protection? One cannot answer this question entirely with specifics, but it is possible to recommend that, in general, a firewall have the following features or attributes:

- The firewall should be able to support a "deny all services except those specifically permitted" design policy, even if that is not the policy used.
- The firewall should support your security policy, not impose one.
- The firewall should be flexible; it should be able to accommodate new services and needs if the security policy of the organization changes.
- The firewall should contain advanced authentication measures or should contain the hooks for installing advanced authentication measures.

4.1.4 Dial-in/out Policy

A useful feature for authorized users is to have remote access to the systems when these users are not on site. A dial-in capability allows them to access systems from locations where Internet access is not available. However as discussed in section 2.3.2, dial-in capabilities add another avenue for intruder access.

Authorized users may also wish to have a dial-out capability to access those systems that cannot be reached through the Internet. These users need to recognize the vulnerabilities they may be creating if they are careless with modem access. A dial-out capability may easily become a dial-in capability if proper precautions are not taken.

The dial-in and dial-out capabilities should be considered in the design of the firewall and incorporated into it. Forcing outside users to go through the advanced authentication of the firewall should be strongly reflected in policy. Policy can also prohibit the use of unauthorized modems attached to host systems and personal computers at the site if the modem capability is offered through the firewall. A strong policy and effective modem service may limit the number of unauthorized modems throughout the site, thus limiting this dangerous vulnerability as well.

4.1.5 Remote Network Connections

In addition to dial-in/dial-out connections, the use of Serial Line IP (SLIP) and Point-to-Point Protocol (PPP) connections need to be considered as part of the policy. Users could use SLIP or PPP to create new network connections into a site protected by a firewall. Such a connection is potentially a backdoor around the firewall, and may be an even larger backdoor than a simple dial-in connection.

Section 3 provided several examples for locating dial-in capability such that dial-in connections would pass first through the firewall. This sort of arrangement could be used as well for SLIP and PPP connections, however this would need to be set forth in policy. As usual, the policy would have to be very strong with regard to these connections.

4.1.6 Information Server Policy

A site that is providing public access to an information server must incorporate this access into the firewall design. While the information server itself creates specific security concerns, the information server should not become a vulnerability to the security of the protected site. Policy should reflect the philosophy that the security of the site will not be compromised in order to provide an information service.

with NFS are acceptable to the organization, it may require changing the design policy to the less secure approach of permitting all services except those specifically denied and passing NFS through the firewall to site systems. Or, it may require obtaining a firewall that can locate the systems that require NFS on a screened subnet, thus preserving the "deny all" design policy for the rest of the site systems. Or, the risks of using NFS may prove too great; NFS would have to be dropped from the list of services to use remotely. The aim of this exercise, then, is to arrive at a service access policy and the firewall design policy.

To assist in this process, the following sections present some common issues that need to be addressed in the policies associated with firewall use.

4.1.2 Flexibility in Policy

Any security policy that deals with Internet access, Internet services, and network access in general, should be flexible. This flexibility must exist for two reasons: the Internet itself is in flux, and the organization's needs may change as the Internet offers new services and methods for doing business. New protocols and services are emerging on the Internet, which offers more benefits to organizations using the Internet, but may also result in new security concerns. Thus, a policy needs to be able to reflect and incorporate these new concerns. The other reason for the flexibility is that the risk of the organization also does not remain static. The change in risk may be a reflection of major changes such as new responsibilities being assigned to the organization, or smaller changes such as a network configuration change.

4.1.3 Remote User Advanced Authentication Policy

Remote users are those who originate connections to site system from elsewhere on the Internet. These connections could come from any location on the Internet, from dial-in lines, or from authorized users on travel or working from home. Regardless, all such connections should use the advanced authentication service of the firewall to access systems at the site. Policy should reflect that remote users may not access systems through unauthorized modems placed behind the firewall. There must be no exceptions to this policy, as it may take only one captured password or one uncontrolled modem line to enable a backdoor around the firewall.

Such a policy has its drawbacks: increased user training for using advanced authentication measures, increased expense if remote users must be supplied with authentication tokens or smartcards, and increased overhead in administering remote access. But, it does not make sense to install a firewall and at the same time not control remote access.

implemented, such as no inbound access and full outbound access or restricted inbound access and restricted outbound access. The firewall design policy determines to a large degree the service access policy: the more robust the firewall design policy, the more stringent the service access policy. Thus, the firewall design policy needs to be decided upon first.

As explained in section 2.4.1, the firewall design policy is generally to deny all services except those that are explicitly permitted or to permit all services except those that are explicitly denied. The former is more secure and is therefore preferred, but it is also more stringent and causes fewer services to be permitted by the service access policy.

Chapter 3 provided several firewall examples, and showed that certain firewalls can implement either design policy whereas one, the dual-homed gateway, is inherently a "deny all" firewall. However, the examples also showed that systems needing certain services that shouldn't be passed through the firewalls could be located on screened subnets separate from other site systems. The point here is that depending on security and flexibility requirements, certain types of firewalls are more appropriate than others. This shows also the importance of choosing a policy first before implementing the firewall; doing the opposite could result in a clumsy fit.

To arrive at a firewall design policy and then ultimately a firewall system that implements the policy, NIST recommends that the firewall design policy start with the most secure, i.e., deny all services except those that are explicitly permitted. The policy designer then should understand and document the following:

- which Internet services the organization plans to use, e.g., TELNET, Mosaic, and NFS,
- where the services will be used, e.g., on a local basis, across the Internet, dial-in from home, or from remote organizations,
- additional needs, such as encryption or dial-in support,
- what are the risks associated with providing these services and access,
- what is the cost in terms of controls and impact on network usability to provide protection, and
- assumptions about security versus usability: does security win out if a particular service is too risky or too expensive to secure.

The creation of these items is straightforward, however at the same time highly iterative. For example, a site may wish to use NFS across two remote sites, however the "deny all" design policy may not permit NFS (as explained in sec. 2.4.1). If the risks associated

4

Next Steps

Up to this point, this document has provided a basic vocabulary of threats and risks associated with the Internet, how Internet firewalls can be used to address those problems, and some examples of firewall implementations. This chapter provides basic guidance on designing a network service policy and choosing a firewall design policy, and then discusses next steps in obtaining a firewall. It closes with a discussion of issues involved in maintaining a firewall and other steps for improving overall network security. The discussion is brief and serves only to raise issues; readers are urged to consult more complete discussions such as [RFC1244] and [Ches94], and specific examples of policies such as in [Avol94].

4.1 Firewall Policy

Policy was discussed in 2.4.1 in terms of a *service access policy* and a *firewall design policy*. This section discusses these policies in relationship to overall site policy, and offers guidance on how to identify needs, risks, and then policies.

Policy decisions regarding the use of firewall technology should be made in conjunction with the policy decisions needed to secure the whole site. This includes decisions concerning host systems security, dial-in access, off-site Internet access, protection of information off-site, data communications security and others. A stand-alone policy concerning only the firewall is not effective; it needs to be incorporated into a strong site security policy. Refer to [RFC1244] for information on creating a site security policy geared towards the needs of Internet sites.

4.1.1 Steps in Creating a Service Access Policy

A firewall is a direct implementation of the network service access and design policies, as discussed in section 2.4.1. There are a number of service access policies that may be

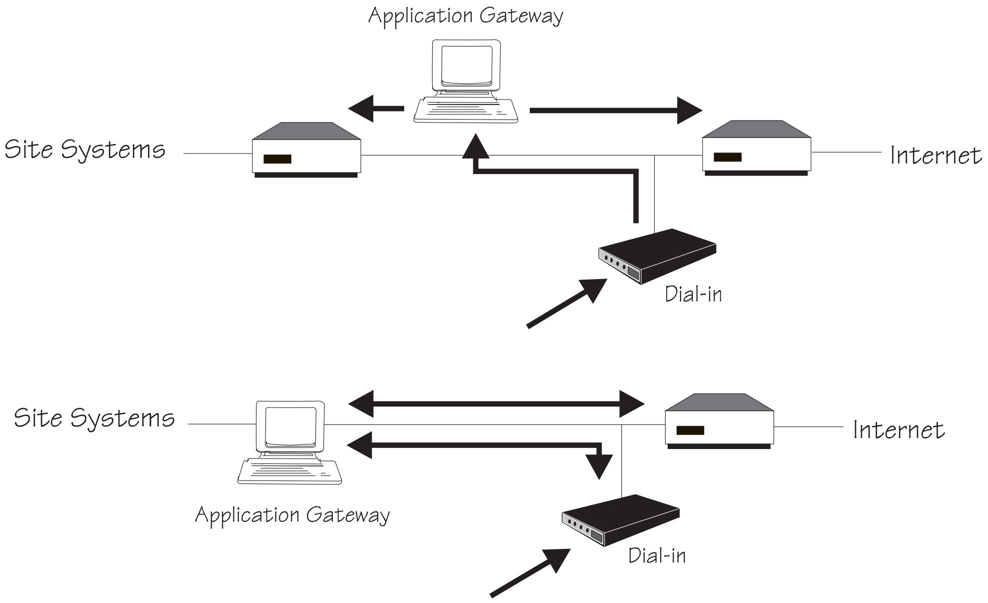

Figure 3.6: Modem Pool Placement with Screened Subnet and Dual-Homed Firewalls.

With the dual-homed gateway and screened subnet firewalls, the router connected to the Internet would prevent routing between Internet systems and the modem pool. With the screened subnet firewall, the router connected to the site would prevent routing between site systems and the modem pool; with the dual-homed gateway firewall, the application gateway would prevent the routing. Users dialing into the modem pool could connect to site systems or the Internet only by connecting to the application gateway, which would use advanced authentication measures.

If a site uses any of these measures to protect dial-in access, it must rigidly enforce a policy that prevents any users from connecting modems elsewhere on the protected subnet. Even if the modems contain security features, this adds more complexity to the firewall protection scheme and adds another "weak link" to the chain.

protection provided by the firewall. A much better method for handling modems is to concentrate them into a modem pool, and then secure connections from that pool.

The modem pool likely would consist of modems connected to a terminal server, which is a specialized computer designed for connecting modems to a network. A dial-in user connects to the terminal server, and then connects (e.g., telnets) from there to other host systems. Some terminal servers provide security features that can restrict connections to specific systems, or require users to authenticate using an authentication token. Alternatively, the terminal server can be a host system with modems connected to it.

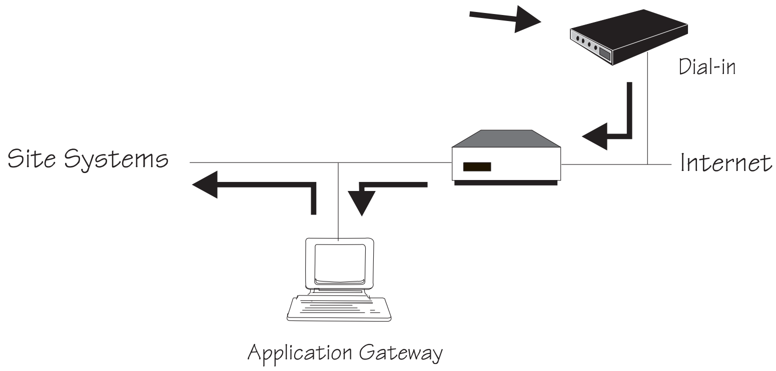

Figure 3.5: Modem Pool Placement with Screened Host Firewall.

Figure 3.5 shows a modem pool located on the Internet side of the screened host firewall. Since the connections from modems need to be treated with the same suspicion as connections from the Internet, locating the modem pool on the outside of the firewall forces the modem connections to pass through the firewall.

The application gateway's advanced authentication measures can be used then to authenticate users who connect from modems as well as from the Internet. The packet filtering router could be used to prevent inside systems from connecting directly to the modem pool.

A disadvantage to this, though, is that the modem pool is connected directly to the Internet and thus more exposed to attack. If an intruder managed to penetrate the modem pool, the intruder might use it as a basis for connecting to and attacking other Internet systems. Thus, a terminal server with security features to reject dial-in connections to any system but the application gateway should be used.

The dual-homed gateway and screened subnet firewalls provide a more secure method for handling modem pools. In figure 3.6, the terminal server gets located on the inner, screened subnet, where access to and from the modem pool can be carefully controlled by the routers and application gateways. The router on the Internet side protects the modem pool from any direct Internet access except from the application gateway.

The two routers provide redundancy in that an attacker would have to subvert *both* routers to reach site systems directly. The application gateway, e-mail server, and information server could be set up such that they would be the only systems "known" from the Internet; no other system name need be known or used in a DNS database that would be accessible to outside systems. The application gateway can house advanced authentication software to authenticate all inbound connections. It is, obviously, more involved to configure, however the use of separate systems for application gateways and packet filters keeps the configuration more simple and manageable.

The screened subnet firewall, like the screened host firewall, can be made more flexible by permitting certain "trusted" services to pass between the Internet and the site systems. However, this flexibility may open the door to exceptions to the policy, thus weakening the effect of the firewall. In many ways, the dual-homed gateway firewall is more desireable because the policy cannot be weakened (because the dual-homed gateway cannot pass services for which there is no proxy). However, where throughput and flexibility are important, the screened subnet firewall may be more preferable.

As an alternative to passing services directly between the Internet and site systems, one could locate the systems that need these services directly on the screened subnet. For example, a site that does not permit X Windows or NFS traffic between Internet and site systems, but needs to anyway, could locate the systems that need the access on the screened subnet. The systems could still maintain access to site systems by connecting to the application gateway and reconfiguring the inner router as necessary. This is not a perfect solution, but an option for sites that require a high degree of security.

There are two disadvantages to the screened subnet firewall. First, the firewall can be made to pass "trusted" services around the application gateway(s), thereby subverting the policy. This is true also with the screened host firewall, however the screened subnet firewall provides a location to house systems that need direct access to those services. With the screened host firewall, the "trusted" services that get passed around the application gateway end up being in contact with site systems. The second disadvantage is that more emphasis is placed on the routers for providing security. As noted, packet filtering routers are sometimes quite complex to configure and mistakes could open the entire site to security holes.

[Ran93] and [Ches94] provide more details on screened subnet firewalls.

3.5 Integrating Modem Pools with Firewalls

Many sites permit dial-in access to modems located at various points throughout the site. As discussed in section 2.3.2, this is a potential backdoor and could negate all the

systems). The router would be used as well to block packets such as NFS, NIS, or any other vulnerable protocols that do not need to pass to or from hosts on the screened subnet.

The inner router passes traffic to and from systems on the screened subnet according to the following rules:

- application traffic from the application gateway to site systems gets routed,
- e-mail traffic from the e-mail server to site systems gets routed,
- application traffic to the application gateway from site systems get routed,
- e-mail traffic from site systems to the e-mail server gets routed,
- ftp, gopher, etc., traffic from site systems to the information server gets routed,
- all other traffic gets rejected.

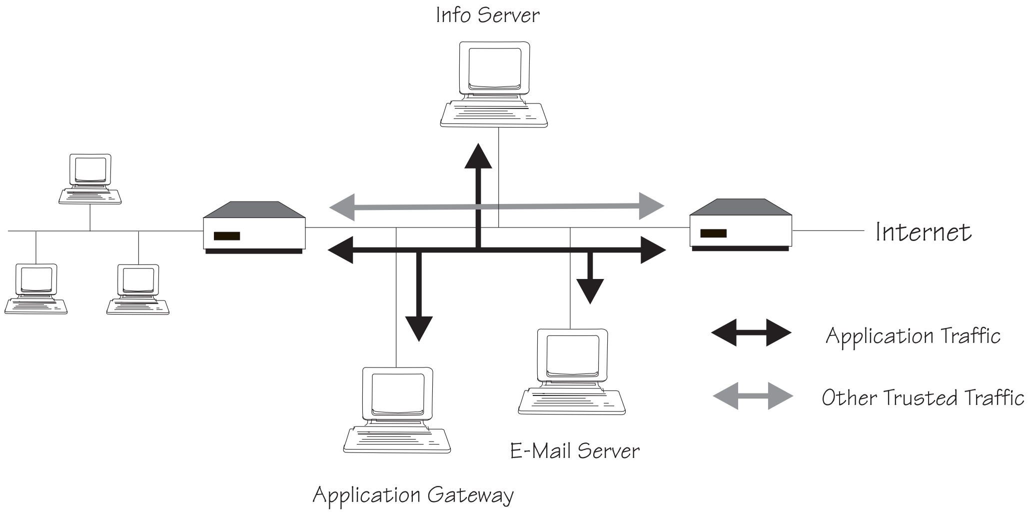

Figure 3.4: Screened Subnet Firewall with Additional Systems.

Thus, no site system is directly reachable from the Internet and vice versa, as with the dual-homed gateway firewall. A big difference, though, is that the routers are used to direct traffic to specific systems, thereby eliminating the need for the application gateway to be dual-homed. Greater throughput can be achieved, then, if a router is used as the gateway to the protected subnet. Consequently, the screened subnet firewall may be more appropriate for sites with large amounts of traffic or sites that need very high-speed traffic.

since the router needs to limit application traffic only to the application gateway, the ruleset may not be as complex as for a typical site using a packet filtering firewall (which may restrict application traffic to multiple systems).

The second disadvantage is that the flexibility opens up the possibility that the policy can be violated (as with the packet filtering firewall). This is less of a problem than with the dual-homed gateway firewall, since it is technically impossible to pass traffic through the dual-homed gateway unless there is a corresponding proxy service. Again, a strong policy is essential.

[Garf92], [Ran93], and [Ches94] provide more details on screened host firewalls.

3.4 Screened Subnet Firewall

The screened subnet firewall is a variation of the dual-homed gateway and screened host firewalls. It can be used to locate each component of the firewall on a separate system, thereby achieving greater throughput and flexibility, although at some cost to simplicity. But, each component system of the firewall needs to implement only a specific task, making the systems less complex to configure.

In figure 3.4, two routers are used to create an inner, *screened* subnet. This subnet (sometimes referred to in other literature as the "DMZ") houses the application gateway, however it could also house information servers, modem pools, and other systems that require carefully-controlled access. The router shown as the connection point to the Internet would route traffic according to the following rules:

- application traffic from the application gateway to Internet systems gets routed,
- e-mail traffic from the e-mail server to Internet sites gets routed,
- application traffic from Internet sites to the application gateway gets routed,
- e-mail traffic from Internet sites to the e-mail server gets routed,
- ftp, gopher, etc., traffic from Internet sites to the information server gets routed, and
- all other traffic gets rejected.

The outer router restricts Internet access to specific systems on the screened subnet, and blocks all other traffic to the Internet originating from systems that should not be originating connections (such as the modem pool, the information server, and site

screens inherently dangerous protocols from reaching the application gateway and site systems. It rejects (or accepts) application traffic according to the following rules:

- application traffic from Internet sites to the application gateway gets routed,
- all other traffic from Internet sites gets rejected, and
- the router rejects any application traffic originating from the inside unless it came from the application gateway.

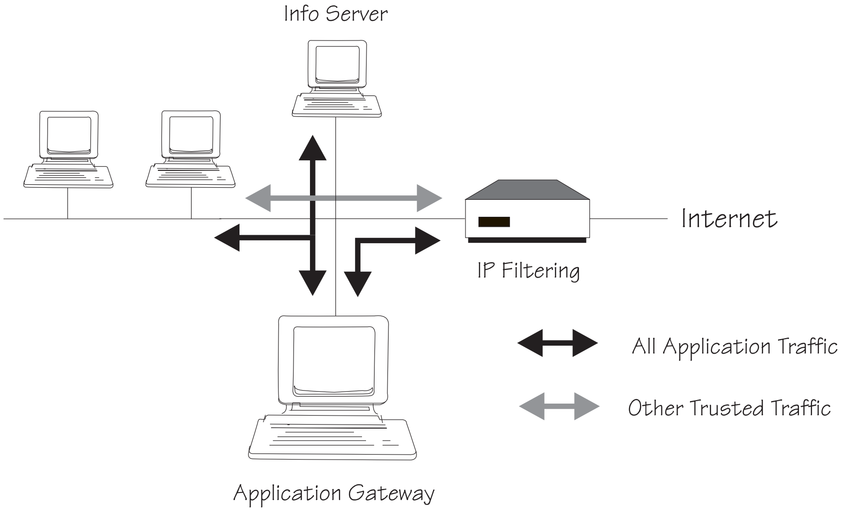

Figure 3.3: Screened Host Firewall.

Unlike the dual-homed gateway firewall, the application gateway needs only one network interface and does not require a separate subnet between the application gateway and the router. This permits the firewall to be made more flexible but perhaps less secure by permitting the router to pass certain trusted services "around" the application gateway and directly to site systems. The trusted services might be those for which proxy services don't exist, and might be trusted in the sense that the risk of using the services has been considered and found acceptable. For example, less-risky services such as NTP could be permitted to pass through the router to site systems. If the site systems require DNS access to Internet systems, DNS could be permitted to site systems. In this configuration, the firewall could implement a mixture of the two design policies, the proportions of which depend on how many and what types of services are routed directly to site systems.

The additional flexibility of the screened host firewall is cause for two concerns. First, there are now two systems, the router and the application gateway, that need to be configured carefully. As noted before, packet filtering router rules can be complex to configure, difficult to test, and prone to mistakes that lead to holes through the router. However,

in this chapter, provides the ability to segregate traffic concerned with an information server from other traffic to and from the site. An information server could be located on the subnet between the gateway and the router, as shown in figure 3.2. Assuming that the gateway provides the appropriate proxy services for the information server (e.g., ftp, gopher, or http), the router can prevent direct Internet access to the firewall and force access to go through the firewall. If direct access is permitted to the server (which is the less secure alternative), then the server's name and IP address can be advertised by DNS. Locating the information server there also adds to the security of the site, as any intruder penetration of the information server would still be prevented from reaching site systems by the dual-homed gateway.

The inflexibility of the dual-homed gateway could be a disadvantage to some sites. Since all services are blocked except those for which proxies exist, access to other services cannot be opened up; systems that require the access would need to be placed on the Internet side of the gateway. However, a router could be used as shown in figure 3.2 to create a subnet between the gateway and the router, and the systems that require extra services could be located there (this is discussed more in section 3.4 with screened subnet firewalls).

Another important consideration is that the security of the host system used for the firewall must be very secure, as the use of any vulnerable services or techniques on the host could lead to break-ins. If the firewall is compromised, an intruder could potentially subvert the firewall and perform some activity such as to re-enable IP routing.

[Garf92], [Ran93], and [Ches94] discuss advantages and disadvantages of dual-homed gateways used as firewalls.

3.3 Screened Host Firewall

The screened host firewall (fig. 3.3) is a more flexible firewall than the dual-homed gateway firewall, however the flexibility is achieved with some cost to security. The screened host firewall is often appropriate for sites that need more flexibility than that provided by the dual-homed gateway firewall.

The screened host firewall combines a packet-filtering router with an application gateway located on the protected subnet side of the router.[6] The application gateway needs only one network interface. The application gateway's proxy services would pass TELNET, FTP, and other services for which proxies exist, to site systems. The router filters or

[6]The application gateway could also be located on the Internet side of the router with no apparent loss in security. Locating the application gateway on the outside may help to reinforce the understanding that it is subject to Internet attacks and should not necessarily be trusted.

forwarding capability disabled (i.e., the default condition is that the host can no longer route packets between the two connected networks). In addition, a packet filtering router can be placed at the Internet connection to provide additional protection. This would create an inner, screened subnet that could be used for locating specialized systems such as information servers and modem pools.

Unlike the packet filtering firewall, the dual-homed gateway is a complete block to IP traffic between the Internet and protected site. Services and access is provided by proxy servers on the gateway. It is a simple firewall, yet very secure.[5]

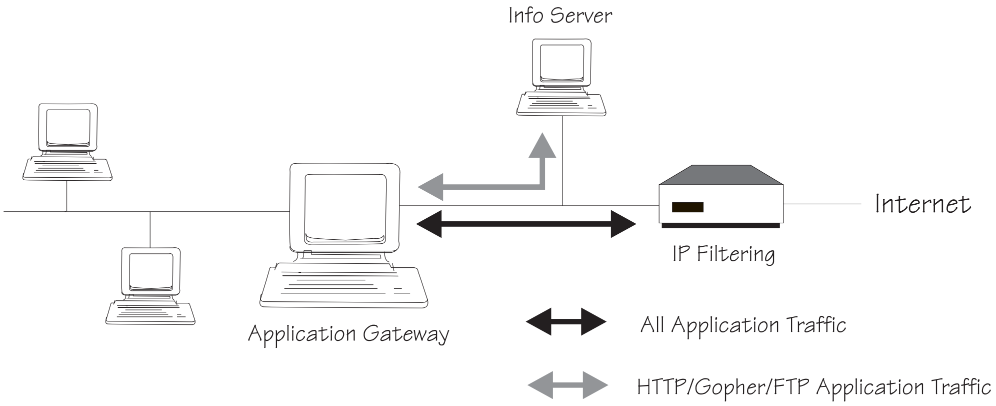

Figure 3.2: Dual-homed Gateway Firewall with Router.

This type of firewall implements the second design policy, i.e., deny all services unless they are specifically permitted, since no services pass except those for which proxies exist. The ability of the host to accept source-routed packets would be disabled, so that no other packets could be passed by the host to the protected subnet. It can be used to achieve a high degree of privacy since routes to the protected subnet need to be known only to the firewall and not to Internet systems (because Internet systems cannot route packets directly to the protected systems). The names and IP addresses of site systems would be hidden from Internet systems, because the firewall would not pass DNS information.

A simple setup for a dual-homed gateway would be to provide proxy services for TELNET and FTP, and centralized e-mail service in which the firewall would accept all site mail and then forward it to site systems. Because it uses a host system, the firewall can house software to require users to use authentication tokens or other advanced authentication measures. The firewall can also log access and log attempts or probes to the system that might indicate intruder activity.

The dual-homed gateway firewall, as well as the screened subnet firewall mentioned later

[5]Some dual-homed gateway firewalls do not use proxy services but require users to have accounts on the gateway for access to the Internet. This firewall is not recommended, as maintaining multiple accounts on a firewall can lead to user mistakes, which can lead to intruder attacks and break-ins.

Basically, one installs a packet filtering router at the Internet (or any subnet) gateway and then configures the packet filtering rules in the router to block or filter protocols and addresses. The site systems usually have direct access to the Internet while all or most access to site systems from the Internet is blocked. However, the router could allow selective access to systems and services, depending on the policy. Usually, inherently-dangerous services such as NIS, NFS, and X Windows are blocked.

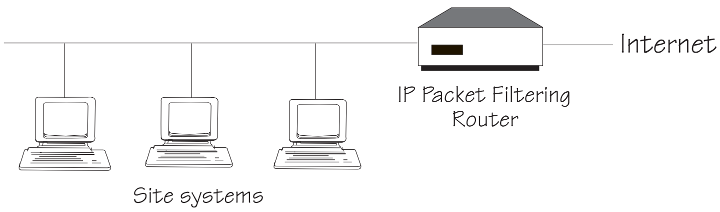

Figure 3.1: Packet Filtering Firewall.

A packet filtering firewall suffers from the same disadvantages as a packet filtering router, however they can become magnified as the security needs of a protected site becomes more complex and stringent. These would include the following:

- there is little or no logging capability, thus an administrator may not easily determine whether the router has been compromised or is under attack,
- packet filtering rules are often difficult to test thoroughly, which may leave a site open to untested vulnerabilities,
- if complex filtering rules are required, the filtering rules may become unmanageable, and
- each host directly accessible from the Internet will require its own copy of advanced authentication measures.

A packet filtering router can implement either of the design policies discussed in section 2.4.1. However, if the router does not filter on source port or filter on inbound as well as outbound packets, it may be more difficult to implement the second policy, i.e., deny everything unless specifically permitted. If the goal is to implement the second policy, a router that provides the most flexibility in the filtering strategy is desirable. Again, see [Chap92] as well as [Ches94] for more information.

3.2 Dual-homed Gateway Firewall

The dual-homed gateway (fig. 3.2) is a better alternative to packet filtering router firewalls. It consists of a host system with two network interfaces, and with the host's IP

3

Putting the Pieces Together: Firewall Examples

Now that the basic components of firewalls have been examined, some examples of different firewall configurations are provided to give a more concrete understanding of firewall implementation. The firewall examples shown here are:

- **Packet Filtering Firewall**,
- **Dual-homed Gateway Firewall**,
- **Screened Host Firewall**, and
- **Screened Subnet Firewall**.

Additionally, a section is provided that discusses methods for integrating dial-in modem access with firewalls. The examples are based loosely on [Ran93], which provides concise but detailed guidance on firewall definition and design. In the examples, assumptions about policy are kept to a minimum, but policy issues that affect the firewall design are pointed out where appropriate. Readers should note that there are many other types of firewalls that are not illustrated here; their absence does not indicate that they are less secure, only that it is impractical to illustrate every potential design. The examples shown here were chosen primarily because they are covered by other literature in more detail and thus serve well as a basis for more study.

3.1 Packet Filtering Firewall

The packet filtering firewall (fig. 3.1) is perhaps most common and easiest to employ for small, uncomplicated sites. However, it suffers from a number of disadvantages and is less desirable as a firewall than the other example firewalls discussed in this chapter.

simply passes bytes between the systems. Another example of a circuit-level gateway would be for NNTP, in which the NNTP server would connect to the firewall, and then internal systems' NNTP clients would connect to the firewall. The firewall would, again, simply pass bytes.

perform additional steps as necessary such as querying for a one-time password. User behavior stays the same, however at the price of requiring a modified client on each system.

In addition to TELNET, application gateways are used generally for FTP and e-mail, as well as for X Windows and some other services. Some FTP application gateways include the capability to deny *put* and *get* command to specific hosts. For example, an outside user who has established an FTP session (via the FTP application gateway) to an internal system such as an anonymous FTP server might try to upload files to the server. The application gateway can filter the FTP protocol and deny all *puts* to the anonymous FTP server; this would ensure that nothing can be uploaded to the server and would provide a higher degree of assurance than relying only on file permissions at the anonymous FTP server to be set correctly.[4]

An e-mail application gateway serves to centralize e-mail collection and distribution to internal hosts and users. To outside users, all internal users would have e-mail addresses of the form:

user@emailhost

where *emailhost* is the name of the e-mail gateway. The gateway would accept mail from outside users and then forward mail along to other internal systems as necessary. Users sending e-mail from internal systems could send it directly from their hosts, or in the case where internal system names are not known outside the protected subnet, the mail would be sent to the application gateway, which could then forward the mail to the destination host. Some e-mail gateways use a more secure version of the *sendmail* program to accept e-mail.

Circuit-Level Gateways

[Ches94] defines another firewall component that other authors sometimes include under the category of application gateway. A *circuit-level gateway* relays TCP connections but does no extra processing or filtering of the protocol. For example, the TELNET application gateway example provided here would be an example of a circuit-level gateway, since once the connection between the source and destination is established, the firewall

[4] Some sites have instituted policies that deny put and get commands in certain directions; having a firewall that can filter FTP commands is especially useful in such a situation. Some sites have disallowed get commands outbound, thus no users could retrieve information or software from outside sources. Other sites have disallowed put commands outbound, thus no users could store information on FTP servers external to the site. More common has been to allow no put commands inbound, thus no external users can write to FTP servers at the site.

This example points out several benefits to using proxy services. First, proxy services allow only those services through for which there is a proxy. In other words, if an application gateway contains proxies for FTP and TELNET, then only FTP and TELNET may be allowed into the protected subnet, and all other services are completely blocked. For some sites, this degree of security is important, as it guarantees that only those services that are deemed “trustworthy” are allowed through the firewall. It also prevents other untrusted services from being implemented behind the backs of the firewall administrators.

Another benefit to using proxy services is that the protocol can be filtered. Some firewalls, for example, can filter FTP connections and deny use of the FTP *put* command, which is useful if one wants to guarantee that users cannot write to, say, an anonymous FTP server.

Application gateways have a number of general advantages over the default mode of permitting application traffic directly to internal hosts. These include:

- **information hiding**, in which the names of internal systems need not necessarily be made known via DNS to outside systems, since the application gateway may be the only host whose name must be made known to outside systems,
- **robust authentication and logging**, in which the application traffic can be pre-authenticated before it reaches internal hosts and can be logged more effectively than if logged with standard host logging,
- **cost-effectiveness**, because third-party software or hardware for authentication or logging need be located only at the application gateway, and
- **less-complex filtering rules**, in which the rules at the packet filtering router will be less complex than they would if the router needed to filter application traffic and direct it to a number of specific systems. The router need only allow application traffic destined for the application gateway and reject the rest.

A disadvantage of application gateways is that, in the case of client-server protocols such as TELNET, two steps are required to connect inbound or outbound. Some application gateways require modified clients, which can be viewed as a disadvantage or an advantage, depending on whether the modified clients make it easier to use the firewall. A TELNET application gateway would not necessarily require a modified TELNET client, however it would require a modification in user behavior: the user has to connect (but not login) to the firewall as opposed to connecting directly to the host. But a modified TELNET client could make the firewall transparent by permitting a user to specify the destination system (as opposed to the firewall) in the TELNET command. The firewall would serve as the route to the destination system and thereby intercept the connection, and then

2.4.4 Application Gateways

To counter some of the weaknesses associated with packet filtering routers, firewalls need to use software applications to forward and filter connections for services such as TELNET and FTP. Such an application is referred to as a *proxy service*, while the host running the proxy service is referred to as an *application gateway*. Application gateways and packet filtering routers can be combined to provide higher levels of security and flexibility than if either were used alone.

As an example, consider a site that blocks all incoming TELNET and FTP connections using a packet filtering router. The router allows TELNET and FTP packets to go to one host only, the TELNET/FTP application gateway. A user who wishes to connect inbound to a site system would have to connect first to the application gateway, and then to the destination host, as follows:

1. a user first telnets to the application gateway and enters the name of an internal host,
2. the gateway checks the user's source IP address and accepts or rejects it according to any access criteria in place,
3. the user may need to authenticate herself (possibly using a one-time password device),
4. the proxy service creates a TELNET connection between the gateway and the internal host,
5. the proxy service then passes bytes between the two connections, and
6. the application gateway logs the connection.

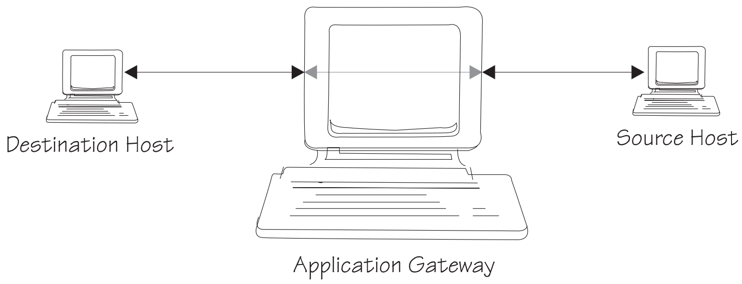

Figure 2.4: Virtual Connection Implemented by an Application Gateway and Proxy Services.

port would be 25, the port that the SMTP server "listens" at. The server would return packets with source port of 25 and destination port equal to the randomly-chosen port at the client. If a site permits both inbound and outbound SMTP connections, the router must allow destination ports and source ports > 1023 in both directions. If the router can filter on source port, it can block all packets coming into the site that have a destination port > 1023 and a source port *other than 25*. Without the ability to filter on source port, the router must permit connections that use source *and* destination ports > 1024. Users could conceivably run servers at ports > 1023 and thus get "around" the filtering policy (i.e., a site system's telnet server that normally listens at port 23 could be told to listen at port 9876 instead; users on the Internet could then telnet to this server even if the router blocks destination port 23).

Another problem is that a number of RPC (Remote Procedure Call) services are very difficult to filter effectively because the associated servers listen at ports that are assigned randomly at system startup. A service known as *portmapper* maps initial calls to RPC services to the assigned service numbers, but there is no such equivalent for a packet filtering router. Since the router cannot be told which ports the services reside at, it isn't possible to block completely these services unless one blocks all UDP packets (RPC services mostly use UDP). Blocking all UDP would block potentially necessary services such as DNS. Thus, blocking RPC results in a dilemma.

Packet filtering routers with more than two interfaces sometimes do not have the capability to filter packets according to which interface the packets arrived at and which interface the packet is bound for. Filtering inbound and outbound packets simplifies the packet filtering rules and permits the router to more easily determine whether an IP address is valid or being spoofed. Routers without this capability offer more impediments to implementing filtering strategies.

Related to this, packet filtering routers can implement both of the design policies discussed in section 2.4.1. A ruleset that is less flexible, i.e., that does not filter on source port or on inbound *and* outbound interfaces, reduces the ability of the router to implement the second and more stringent policy, deny all services except those expressly permitted, without having to curtail the types of services permitted through the router. For example, problematic services such as those that are RPC-based become even more difficult to filter with a less-flexible ruleset; no filtering on source port forces one to permit connections between ports > 1023. With a less-flexible ruleset, the router is less able to express a stringent policy, and the first policy, permit all services except those expressly permitted, is usually followed.

Readers are advised to consult [Chap92], which provides a concise overview of packet filtering and associated problems. While packet filtering is a vital and important tool, it is very important to understand the problems and how they can be addressed.

- **UUCP**, port 540, UNIX-to-UNIX CoPy, if improperly configured can be used for unauthorized access,
- **NNTP**, port 119, Network News Transfer Protocol, for accessing and reading network news, and
- **gopher, http (for Mosaic)**, ports 70 and 80, information servers and client programs for gopher and WWW clients, should be restricted to an application gateway that contains proxy services.

While some of these services such as TELNET or FTP are inherently risky, blocking access to these services completely may be too drastic a policy for many sites. Not all systems, though, generally require access to all services. For example, restricting TELNET or FTP access *from* the Internet to only those systems that require the access can improve security at no cost to user convenience. Services such as NNTP may seem to pose little threat, but restricting these services to only those systems that need them helps to create a *cleaner* network environment and reduces the likelihood of exploitation from yet-to-be-discovered vulnerabilities and threats.

Problems with Packet Filtering Routers

Packet filtering routers suffer from a number of weaknesses, as described in [Chap92]. Packet filtering rules are complex to specify and usually no testing facility exists for verifying the correctness of the rules (other than by exhaustive testing by hand). Some routers do not provide any logging capability, so that if a router's rules still let dangerous packets through, the packets may not be detected until a break-in has occurred.

Often times, exceptions to rules need to be made to allow certain types of access that normally would be blocked. But, exceptions to packet filtering rules sometimes can make the filtering rules so complex as to be unmanageable. For example, it is relatively straightforward to specify a rule to block all inbound connections to port 23 (the TELNET server). If exceptions are made, i.e., if certain site systems need to accept TELNET connections directly, then a rule for each system must be added. Sometimes the addition of certain rules may complicate the entire filtering scheme. As noted previously, testing a complex set of rules for correctness may be so difficult as to be impractical.

Some packet filtering routers do not filter on the TCP/UDP source port, which can make the filtering ruleset more complex and can open up "holes" in the filtering scheme. [Chap92] describes such a problem with sites that wish to allow inbound and outbound SMTP connections. As described in section 1.2.5, TCP connections include a source and destination port. In the case of a system initiating an SMTP connection to a server, the source port would be a randomly chosen port at or above 1024 and the destination

(129.6.48.254 is the only NNTP server that the site should receive news from, thus access to the site for NNTP is restricted to only that system). The fifth rule permits NTP traffic, which uses UDP as opposed to TCP, from any source to any destination address at the site. Finally, the sixth rule denies all other packets - if this rule weren't present, the router may or may not deny all subsequent packets. This is a very basic example of packet filtering. Actual rules permit more complex filtering and greater flexibility.

Which Protocols to Filter

The decision to filter certain protocols and fields depends on the network access policy, i.e., which systems should have Internet access and the type of access to permit. The following services are inherently vulnerable to abuse and are usually blocked at a firewall from entering or leaving the site [Chap92], [Garf92]:

- **tftp**, port 69, trivial FTP, used for booting diskless workstations, terminal servers and routers, can also be used to read any file on the system if set up incorrectly,
- **X Windows, OpenWindows**, ports 6000+, port 2000, can leak information from X window displays including all keystrokes,
- **RPC**, port 111, Remote Procedure Call services including NIS and NFS, which can be used to steal system information such as passwords and read and write to files, and
- **rlogin, rsh, and rexec**, ports 513, 514, and 512, services that if improperly configured can permit unauthorized access to accounts and commands.

Other services, whether inherently dangerous or not, are usually filtered and possibly restricted to only those systems that need them. These would include:

- **TELNET**, port 23, often restricted to only certain systems,
- **FTP**, ports 20 and 21, like TELNET, often restricted to only certain systems,
- **SMTP**, port 25, often restricted to a central e-mail server,
- **RIP**, port 520, routing information protocol, can be spoofed to redirect packet routing,
- **DNS**, port 53, domain names service zone transfers, contains names of hosts and information about hosts that could be helpful to attackers, could be spoofed,

system. With filtering on TCP or UDP ports, this policy can be implemented in a straightforward fashion by a packet filtering router or by a host with packet filtering capability.

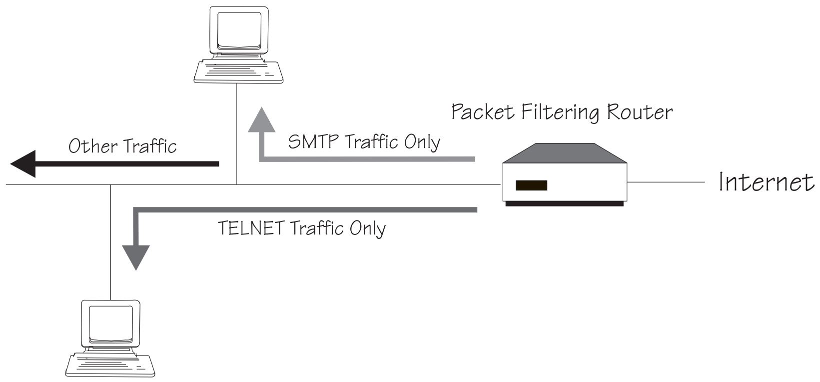

Figure 2.3: Representation of Packet Filtering on TELNET and SMTP.

As an example of packet filtering, consider a policy to allow only certain connections to a network of address 123.4.*.*. TELNET connections will be allowed to only one host, 123.4.5.6, which may be the site's TELNET application gateway, and SMTP connections will be allowed to two hosts, 123.4.5.7 and 123.4.5.8, which may be the site's two electronic mail gateways. NNTP (Network News Transfer Protocol) is allowed only from the site's NNTP feed system, 129.6.48.254, and only to the site's NNTP server, 123.4.5.9, and NTP (Network Time Protocol) is allowed to all hosts. All other services and packets are to be blocked. An example of the ruleset would be as follows:

Type	**Source Addr**	**Dest Addr**	**Source Port**	**Dest Port**	**Action**
tcp	*	123.4.5.6	> 1023	23	permit
tcp	*	123.4.5.7	> 1023	25	permit
tcp	*	123.4.5.8	> 1023	25	permit
tcp	129.6.48.254	123.4.5.9	> 1023	119	permit
udp	*	123.4.*.*	> 1023	123	permit
*	*	*	*	*	deny

The first rule allows TCP packets from any source address and port *greater than 1023* on the Internet to the destination address of 123.4.5.6 and port of 23 at the site. Port 23 is the port associated with the TELNET server, and all TELNET clients should have unprivileged source ports of 1024 or higher. The second and third rules work in a similar fashion, except packets to destination addresses 123.4.5.7 and 123.4.5.8, and port 25 for SMTP, are permitted. The fourth rule permits packets to the site's NNTP server, but only from source address 129.6.48.254 to destination address 123.4.5.9 and port 119

permitting access, however these passwords would be immune from exploitation, even if the passwords are monitored, as long as the advanced authentication measures and other firewall components prevent intruders from penetrating or bypassing the firewall.

Sections 2.4.4 and 3 contain more information on using advanced authentication measures with firewalls. See [NIST94b] for more information on using advanced authentication measures with hosts.

2.4.3 Packet Filtering

IP packet filtering is done usually using a *packet filtering router* designed for filtering packets as they pass between the router's interfaces. A packet filtering router usually can filter IP packets based on some or all of the following fields:

- **source IP address**,
- **destination IP address**,
- **TCP/UDP source port**, and
- **TCP/UDP destination port**.

Not all packet filtering routers currently filter the source TCP/UDP port, however more vendors are starting to incorporate this capability. Some routers examine which of the router's network interfaces a packet arrived at, and then use this as an additional filtering criterion. Some UNIX hosts provide packet filtering capability, although most do not.

Filtering can be used in a variety of ways to block connections from or to specific hosts or networks, and to block connections to specific ports. A site might wish to block connections from certain addresses, such as from hosts or sites that it considers to be hostile or untrustworthy. Alternatively, a site may wish to block connections from *all* addresses external to the site (with certain exceptions, such as with SMTP for receiving e-mail).

Adding TCP or UDP port filtering to IP address filtering results in a great deal of flexibility. Recall from Chapter 1 that servers such as the TELNET daemon *reside* usually at specific ports, such as port 23 for TELNET. If a firewall can block TCP or UDP connections to or from specific ports, then one can implement policies that call for certain types of connections to be made to specific hosts, but not other hosts. For example, a site may wish to block all incoming connections to all hosts except for several firewalls-related systems. At those systems, the site may wish to allow only specific services, such as SMTP for one system and TELNET or FTP connections to another

passwords. However, even if users follow this advice (and many do not), the fact that intruders can and do monitor the Internet for passwords that are transmitted in the clear has rendered traditional passwords obsolete.

Advanced authentication measures such as smartcards, authentication tokens, biometrics, and software-based mechanisms are designed to counter the weaknesses of traditional passwords. While the authentication techniques vary, they are similar in that the passwords generated by advanced authentication devices cannot be reused by an attacker who has monitored a connection. Given the inherent problems with passwords on the Internet, an Internet-accessible firewall that does not use or does not contain the *hooks* to use advanced authentication makes little sense.

Some of the more popular advanced authentication devices in use today are called *one-time password systems*. A smartcard or authentication token, for example, generates a response that the host system can use in place of a traditional password. Because the token or card works in conjunction with software or hardware on the host, the generated response is unique for every login. The result is a one-time password that, if monitored, cannot be reused by an intruder to gain access to an account. [NIST94a] and [NIST91a] contain more detail on advanced authentication devices and measures.

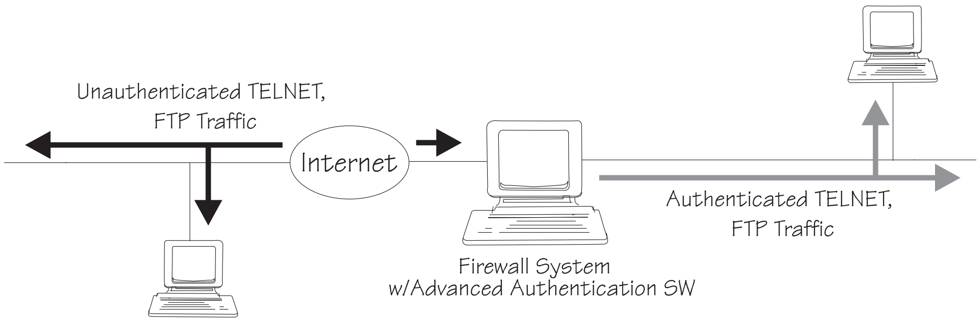

Figure 2.2: Use of Advanced Authentication on a Firewall to Preauthenticate TELNET, FTP Traffic.

Since firewalls can centralize and control site access, the firewall is the logical place for the advanced authentication software or hardware to be located. Although advanced authentication measures could be used at each host, it is more practical and manageable to centralize the measures at the firewall. Figure 2.2 illustrates that a site without a firewall using advanced authentication permits unauthenticated application traffic such as TELNET or FTP directly to site systems. If the hosts do not use advanced authentication, then intruders could attempt to crack passwords or could monitor the network for login sessions that would include the passwords. Figure 2.2 also shows a site with a firewall using advanced authentication, such that TELNET or FTP sessions originating from the Internet to site systems must pass the advanced authentication before being permitted to the site systems. The site systems may still require static passwords before

2. *deny* any service unless it is expressly permitted.

A firewall that implements the first policy allows all services to pass into the site by default, with the exception of those services that the service access policy has identified as disallowed. A firewall that implements the second policy denies all services by default, but then passes those services that have been identified as allowed. This second policy follows the classic access model used in all areas of information security.

The first policy is less desirable, since it offers more avenues for getting around the firewall, e.g., users could access new services currently not denied by the policy (or even addressed by the policy) or run denied services at non-standard TCP/UDP ports that aren't denied by the policy. Certain services such as X Windows, FTP, Archie, and RPC cannot be filtered easily [Chap92], [Ches94], and are better accommodated by a firewall that implements the first policy. The second policy is stronger and safer, but it is more difficult to implement and may impact users more in that certain services such as those just mentioned may have to be blocked or restricted more heavily.

The relationship between the high level service access policy and its lower level counterpart is reflected in the discussion above. This relationship exists because the implementation of the service access policy is so heavily dependent upon the capabilities and limitations of the firewall system, as well as the inherent security problems associated with the wanted Internet services. For example, wanted services defined in the service access policy may have to be denied if the inherent security problems in these services cannot be effectively controlled by the lower level policy and if the security of the network takes precedence over other factors. On the other hand, an organization that is heavily dependent on these services to meet its mission may have to accept higher risk and allow access to these services. This relationship between the service access policy and its lower level counterpart allows for an iterative process in defining both, thus producing the realistic and sound policy initially described.

The service access policy is the most significant component of the four described here. The other three components are used to implement and enforce the policy. (And as noted above, the service access policy should be a reflection of a strong overall organization security policy.) The effectiveness of the firewall system in protecting the network depends on the type of firewall implementation used, the use of proper firewall procedures, and the service access policy.

2.4.2 Advanced Authentication

Sections 1.3, 1.3.1, and 1.3.2 describe incidents on the Internet that have occurred in part due to the weaknesses associated with traditional passwords. For years, users have been advised to choose passwords that would be difficult to guess and to not reveal their

2.4.1 Network Policy

There are two levels of network policy that directly influence the design, installation and use of a firewall system. The higher-level policy is an issue-specific, network access policy that defines those services that will be allowed or explicitly denied from the restricted network, how these services will be used, and the conditions for exceptions to this policy. The lower-level policy describes how the firewall will actually go about restricting the access and filtering the services that were defined in the higher level policy. The following sections describe these policies in brief.

Service Access Policy

The service access policy should focus on Internet-specific use issues as defined above, and perhaps all outside network access (i.e., dial-in policy, and SLIP and PPP connections) as well. This policy should be an extension of an overall organizational policy regarding the protection of information resources in the organization. For a firewall to be successful, the service access policy must be realistic and sound and should be drafted before implementing a firewall. A realistic policy is one that provides a balance between protecting the network from known risks, while still providing users access to network resources. If a firewall system denies or restricts services, it usually requires the strength of the service access policy to prevent the firewall's access controls from being modified on an ad hoc basis. Only a management-backed, sound policy can provide this.

A firewall can implement a number of service access policies, however a typical policy may be to allow no access to a site from the Internet, but allow access from the site to the Internet. Another typical policy would be to allow some access from the Internet, but perhaps only to selected systems such as information servers and e-mail servers. Firewalls often implement service access policies that allow some user access from the Internet to selected internal hosts, but this access would be granted only if necessary and only if it could be combined with advanced authentication.

Firewall Design Policy

The firewall design policy is specific to the firewall. It defines the rules used to implement the service access policy. One cannot design this policy in a vacuum isolated from understanding issues such as firewall capabilities and limitations, and threats and vulnerabilities associated with TCP/IP. Firewalls generally implement one of two basic design policies:

1. *permit* any service unless it is expressly denied, and

- **MBONE** - Multicast IP transmissions (MBONE) for video and voice are encapsulated in other packets; firewalls generally forward the packets without examining the packet contents. MBONE transmissions represent a potential threat if the packets were to contain commands to alter security controls and permit intruders.

- **viruses** - Firewalls do not protect against users downloading virus-infected personal computer programs from Internet archives or transferring such programs in attachments to e-mail. Because these programs can be encoded or compressed in any number of ways, a firewall cannot scan such programs to search for virus signatures with any degree of accuracy. The virus problem still exists and must be handled with other policy and anti-viral controls.

- **throughput** - Firewalls represent a potential bottleneck, since all connections must pass through the firewall and, in some cases, be examined by the firewall. However, this is generally not a problem today, as firewalls can pass data at T1 (1.5 Megabits/second) rates and most Internet sites are at connection rates less than or equal to T1.

- **all eggs in single basket** - A firewall system concentrates security in one spot as opposed to distributing it among systems. A compromise of the firewall could be disastrous to other less-protected systems on the subnet. This weakness can be countered, however, with the argument that lapses and weaknesses in security are more likely to be found as the number of systems in a subnet increase, thereby multiplying the ways in which subnets can be exploited.

Despite these disadvantages, NIST strongly recommends that sites protect their resources with firewalls and other security tools and techniques.

2.4 Firewall Components

The primary components (or aspects) of a firewall are:

- **network policy**,
- **advanced authentication mechanisms**,
- **packet filtering**, and
- **application gateways**.

The following sections describe each of these components more fully.

of network use. For example, a site might depend on using NFS and NIS across major gateways. In such a situation, the relative costs of adding a firewall would need to be compared against the cost of the vulnerabilities associated with not using a firewall, i.e., a risk analysis, and then a decision made on the outcome of the analysis. Other solutions such as Kerberos may be more appropriate, however these solutions carry their own disadvantages as well. [NIST94c] contains more information on Kerberos and other potential solutions.

2.3.2 Large Potential for Back Doors

Secondly, firewalls do not protect against back doors into the site. For example, if unrestricted modem access is still permitted into a site protected by a firewall, attackers could effectively jump around the firewall [Haf91]. Modem speeds are now fast enough to make running SLIP (Serial Line IP) and PPP (Point-to-Point Protocol) practical; a SLIP or PPP connection inside a protected subnet is in essence another network connection and a potential backdoor. Why have a firewall if unrestricted modem access is permitted?

2.3.3 Little Protection from Insider Attacks

Firewalls generally do not provide protection from insider threats. While a firewall may be designed to prevent outsiders from obtaining sensitive data, the firewall does not prevent an insider from copying the data onto a tape and taking it out of the facility. Thus, it is faulty to assume that the existence of a firewall provides protection from insider attacks or attacks in general that do not need to use the firewall. It is perhaps unwise to invest significant resources in a firewall if other avenues for stealing data or attacking systems are neglected.

2.3.4 Other Issues

Other problems or issues with firewalls are as follows:

- **WWW, gopher** - Newer information servers and clients such as those for World Wide Web (WWW), gopher, WAIS, and others were not designed to work well with firewall policies and, due to their newness, are generally considered risky. The potential exists for data-driven attacks, in which data processed by the clients can contain instructions to the clients; the instructions could tell the client to alter access controls and important security-related files on the host.

2.2.5 Logging and Statistics on Network Use, Misuse

If all access to and from the Internet passes through a firewall, the firewall can log accesses and provide valuable statistics about network usage. A firewall, with appropriate *alarms* that sound when suspicious activity occurs can also provide details on whether the firewall and network are being probed or attacked.

It is important to collect network usage statistics and evidence of probing for a number of reasons. Of primary importance is knowing whether the firewall is withstanding probes and attacks, and determining whether the controls on the firewall are adequate. Network usage statistics are also important as input into network requirements studies and risk analysis activities.

2.2.6 Policy Enforcement

Lastly, but perhaps most importantly, a firewall provides the means for implementing and enforcing a network access policy. In effect, a firewall provides access control to users and services. Thus, a network access policy can be enforced by a firewall, whereas without a firewall, such a policy depends entirely on the cooperation of users. A site may be able to depend on its own users for their cooperation, however it cannot nor should not depend on Internet users in general.

2.3 Issues and Problems with Firewalls

Given these benefits to the firewall approach, there are also a number of disadvantages, and there are a number of things that firewalls cannot protect against. A firewall is not by any means a panacea for Internet security problems.

2.3.1 Restricted Access to Desirable Services

The most obvious disadvantage of a firewall is that it may likely block certain services that users want, such as TELNET, FTP, X Windows, NFS, etc. However, these disadvantage are not unique to firewalls; network access could be restricted at the host level as well, depending on a site's security policy. A well-planned security policy that balances security requirements with user needs can help greatly to alleviate problems with reduced access to services.

Some sites may have a topology that does not lend itself to a firewall, or may use services such as NFS in such a manner that using a firewall would require a major restructuring

2.2.2 Controlled Access to Site Systems

A firewall also provides the ability to control access to site systems. For example, some hosts can be made reachable from outside networks, whereas others can be effectively sealed off from unwanted access. A site could prevent outside access to its hosts except for special cases such as mail servers or information servers.

This brings to the fore an access policy that firewalls are particularly adept at enforcing: do not provide access to hosts or services that do not require access. Put differently, why provide access to hosts and services that could be exploited by attackers when the access is not used or required? If, for example, a user requires little or no network access to her desktop workstation, then a firewall can enforce this policy.

2.2.3 Concentrated Security

A firewall can actually be less expensive for an organization in that all or most modified software and additional security software could be located on the firewall systems as opposed to being distributed on many hosts. In particular, one-time password systems and other add-on authentication software could be located at the firewall as opposed to each system that needed to be accessed from the Internet.

Other solutions to network security such as Kerberos [NIST94c] involve modifications at each host system. While Kerberos and other techniques should be considered for their advantages and may be more appropriate than firewalls in certain situations, firewalls tend to be simpler to implement in that only the firewall need run specialized software.

2.2.4 Enhanced Privacy

Privacy is of great concern to certain sites, since what would normally be considered innocuous information might actually contain clues that would be useful to an attacker. Using a firewall, some sites wish to block services such as *finger* and Domain Name Service. *finger* displays information about users such as their last login time, whether they've read mail, and other items. But, *finger* could leak information to attackers about how often a system is used, whether the system has active users connected, and whether the system could be attacked without drawing attention.

Firewalls can also be used to block DNS information about site systems, thus the names and IP addresses of site systems would not be available to Internet hosts. Some sites feel that by blocking this information, they are hiding information that would otherwise be useful to attackers.

the firewall, where they can be examined and evaluated.

A firewall system can be a router, a personal computer, a host, or a collection of hosts, set up specifically to shield a site or subnet from protocols and services that can be abused from hosts outside the subnet. A firewall system is usually located at a higher-level gateway, such as a site's connection to the Internet, however firewall systems can be located at lower-level gateways to provide protection for some smaller collection of hosts or subnets.

2.2 Why Firewalls

The general reasoning behind firewall usage is that without a firewall, a subnet's systems expose themselves to inherently insecure services such as NFS or NIS and to probes and attacks from hosts elsewhere on the network. In a firewall-less environment, network security relies totally on host security and all hosts must, in a sense, cooperate to achieve a uniformly high level of security. The larger the subnet, the less manageable it is to maintain all hosts at the same level of security. As mistakes and lapses in security become more common, break-ins occur not as the result of complex attacks, but because of simple errors in configuration and inadequate passwords.

A firewall approach provides numerous advantages to sites by helping to increase overall host security. The following sections summarize the primary benefits of using a firewall.

2.2.1 Protection from Vulnerable Services

A firewall can greatly improve network security and reduce risks to hosts on the subnet by filtering inherently insecure services. As a result, the subnet network environment is exposed to fewer risks, since only selected protocols will be able to pass through the firewall.

For example, a firewall could prohibit certain vulnerable services such as NFS from entering or leaving a protected subnet. This provides the benefit of preventing the services from being exploited by outside attackers, but at the same time permits the use of these services with greatly reduced risk to exploitation. Services such as NIS or NFS that are particularly useful on a local area network basis can thus be enjoyed and used to reduce the host management burden.

Firewalls can also provide protection from routing-based attacks, such as source routing and attempts to redirect routing paths to compromised sites via ICMP redirects. A firewall could reject all source-routed packets and ICMP redirects and then inform administrators of the incidents.

2

Introduction to Firewalls

A number of the security problems with the Internet discussed in Chapter 1 can be remedied or made less serious through the use of existing and well-known techniques and controls for host security. A firewall can significantly improve the level of site security while at the same time permitting access to vital Internet services. This chapter provides an overview of firewalls, including how they protect against the vulnerabilities described in Chapter 1, what firewalls *don't* protect against, and the components that make up a firewall. This chapter gives special emphasis to the use of advanced authentication and the importance of policy for determining how a firewall will implement a protection scheme.

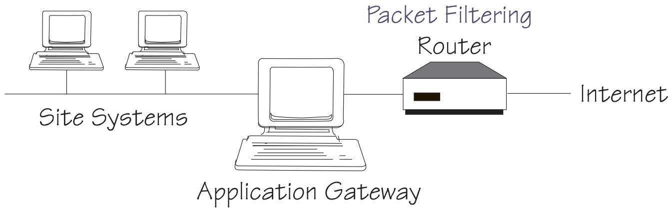

Figure 2.1: Router and Application Gateway Firewall Example.

2.1 The Firewall Concept

Perhaps it is best to describe first what a firewall is *not*: a firewall is not simply a router, host system, or collection of systems that provides security to a network. Rather, a firewall is an approach to security; it helps implement a larger security policy that defines the services and access to be permitted, and it is an implementation of that policy in terms of a network configuration, one or more host systems and routers, and other security measures such as advanced authentication in place of static passwords. The main purpose of a firewall *system* is to control access to or from a protected network (i.e., a site). It implements a network access policy by forcing connections to pass through

gle incident could be comprised of hundreds of break-ins at different sites. It is difficult to draw strong conclusions as to whether the number of incidents and break-ins has remained proportionally the same. Further complicating this is that more people are aware of the existence of incident response teams and may be more likely to report incidents, thus one wonders whether there are more incidents or just more incidents reported.

NIST asserts that the Internet, while a useful and vital network, is at the same time very vulnerable to attacks. Sites that are connected to the Internet face some risk that site systems will be attacked or affected in some form by intruders, and that the risk is significant. The following factors would influence the level of risk:

- the number of systems at the site,
- what services the site uses,
- how interconnected the site is to the Internet,
- the site's profile, or how well-known the site is, and
- how prepared the site is to handle computer security incidents.

The more systems that are connected, obviously the harder it is to control their security. Equally, if a site is connected to the Internet at several points, it likely would be more vulnerable to attacks than a site with a single gateway. At the same time, though, how well prepared a site is, and the degree to which the site *relies* on the Internet, can increase or decrease the risk. A site's high profile could attract more potential intruders who wish to do some harm to the site's image. It should be mentioned, though, that "quiet," less-frequently used sites are also attractive to intruders since they can more easily hide their activity.

NIST asserts that sites that use recommended procedures and controls for increasing computer security have significantly lower risks of attack. Firewalls, combined with one-time passwords that are immune from monitoring or guessing, can increase greatly a site's overall level of security and make using the Internet quite safe. The following chapters contain more detail on firewalls and how they can be used to protect against many of the threats and vulnerabilities mentioned and referenced in this chapter.

to systems. The bugs exist in part because of the complexity of the software and the inability to test it in all the environments in which it must operate. Sometimes the bugs are easily discovered and corrected, other times little can be done except to rewrite the application, which is usually the option of last resort (the sendmail program may be an example of the latter).

1.3.7 Host-based Security Does Not Scale

Host-based security does not scale well: as the number of hosts at a site increases, the ability to ensure that security is at a high level for each host decreases. Given that secure management of just one system can be demanding, managing many such systems could easily result in mistakes and omissions. A contributing factor is that the role of system management is often short-changed and performed in haste. As a result, some systems will be less secure than other systems, and these systems could be the weak links that ultimately will "break" the overall security chain.

If a vulnerability is discovered in networking software, a site that is not protected by a firewall needs to correct the vulnerability on all exposed systems as quickly as possible. As discussed in section 1.3.2, some vulnerabilities have permitted easy access to the UNIX root account; a site with many UNIX hosts would be particularly at risk to intruders in such a situation. Patching vulnerabilities on many systems in a short amount of time may not be practical and, if different versions of the operating system are in use, may not be possible. Such a site would be a "sitting duck" to intruder activity.

1.4 How Vulnerable Are Internet Sites?

As noted in the preceding sections, a number of the TCP and UDP services provide poor levels of security in today's Internet environment. With millions of users connected to the Internet, and governments and industry placing more reliance on Internet availability, the flaws in these services, as well as the availability of source code and tools to automate breaking into systems, can be devastating to sites that suffer break-ins. However, it is difficult to know or assess the true risks of using the Internet and, following, how vulnerable a site is to some form of attack from intruders and related activity. There are no firm statistics.

The Computer Emergency Response Team Coordination Center (CERT/CC) has maintained some base statistics on the number of incidents they have handled since their inception in 1988. The numbers have climbed quite steeply as each year has progressed, however at the same time, the Internet has also grown dramatically. In some cases, CERT counts multiple break-ins of the same pattern as all part of a single incident, thus a sin-

Other services, such as Domain Name Service, can be spoofed, but with more difficulty than electronic mail. These services still represent a threat that needs to be considered when using them.

1.3.5 Flawed LAN Services and Mutually Trusting Hosts

Host systems are difficult and time consuming to manage securely. To ease management demands and to enhance local area networking, some sites use services such as Network Information Services (NIS) and Network File System (NFS). These services can greatly reduce the amount of redundant management by permitting certain databases such as the password files to be managed in a distributed manner and by permitting systems to share files and data. Ironically, these services are inherently insecure and can be exploited to gain access by knowledgeable intruders. If a central server system is compromised, then the other systems trusting the central system could be compromised rather easily.

Some services such as *rlogin* allow for hosts to "trust" each other for the purposes of user convenience and enhanced sharing of systems and devices. If a system is penetrated or spoofed, and that system is trusted by other systems, it is simple for the intruder to then gain access to the other systems. As an example, a user with an account on more than one system can eliminate the need to enter a password at every system by configuring the accounts to trust connections from the user's primary system. When the user uses the *rlogin* command to connect to a host, the destination system will not ask for a password or account name, and the user's connection will be accepted. While this has a positive aspect in that the user's password does not get transmitted and could not be monitored and captured, it has a negative aspect in that if the user's primary account were to be penetrated, the intruder could simply use *rlogin* to penetrate the accounts on the other systems. For this reason, use of mutually-trusting hosts is discouraged [Bel89], [Ches94].

1.3.6 Complex Configuration and Controls

Host system access controls are often complex to configure and test for correctness. As a result, controls that are accidentally misconfigured can result in intruders gaining access. Some major UNIX vendors still ship host systems with access controls configured for maximum (i.e., least secure) access, which can result in unauthorized access if left as is.

A number of security incidents have occurred on the Internet due in part to vulnerabilities discovered by intruders (and subsequently, users, incident response teams, and vendors). Since most modern variants of UNIX derive their networking code from the BSD releases, and since the source code to the BSD releases is widely available, intruders have been able to study the code for bugs and conditions that can be exploited to gain access

an option that can be used to specify a direct route to a destination and return path back to the origination. The route can involve the use of other routers or hosts that normally would not be used to forward packets to the destination. An example of how this can be used such that an attacker's system could masquerade as the trusted client of a particular server is as follows:

1. the attacker would change her host's IP address to match that of the trusted client,
2. the attacker would then construct a source route to the server that specifies the direct path the IP packets should take to the server and should take from the server back to the attacker's host, using the trusted client as the last *hop* in the route to the server,
3. the attacker sends a client request to the server using the source route,
4. the server accepts the client request as if it came directly from the trusted client and returns a reply to the trusted client,
5. the trusted client, using the source route, forwards the packet on to the attacker's host.

Many UNIX hosts accept source routed packets and will pass them on as the source route indicates. Many routers will accept source routed packets as well, whereas some routers can be configured to block source routed packets.

An even simpler method for spoofing a client is to wait until the client system is turned off and then impersonate the client's system. In many organizations, staff members use personal computers and TCP/IP network software to connect to and utilize UNIX hosts as a local area network server. The personal computers often use NFS to obtain access to server directories and files (NFS uses IP addresses only to authenticate clients). An attacker could, after hours, configure a personal computer with the same name and IP address as another's, and then initiate connections to the UNIX host as if it were the "real" client. This is very simple to accomplish and likely would be an insider attack.

Electronic mail on the Internet is particularly easy to spoof and, without enhancements such as digital signatures[NIST94a], generally cannot be trusted. As a brief example, consider the exchange that takes place when Internet hosts exchange mail. The exchange takes place using a simple protocol consisting of ASCII-character commands. An intruder easily could enter these commands by hand by using TELNET to connect directly to a system's Simple Mail Transfer Protocol (SMTP) port. The receiving host trusts that the sending host is who it says it is, thus the origin of the mail can be spoofed easily by entering a sender address that is different from the true address. As a result, any user, without privileges, can falsify or spoof e-mail.

users. The password file can be obtained by simply copying it or via a number of other intruder methods. Once the file is obtained, an intruder can run readily-available password cracking programs against the passwords. If the passwords are weak, e.g., less that 8 characters, English words, etc., they could be cracked and used to gain access into the system.

Another problem with authentication results from some TCP or UDP services being able to authenticate only to the granularity of host addresses and not to specific users. For example, an NFS (UDP) server cannot grant access to a specific user on a host, it must grant access to the entire host. The administrator of a server may trust a specific user on a host and wish to grant access to that user, but the administrator has no control over other users on that host and is thus forced to grant access to all users (or grant no access at all).

1.3.3 Ease of Spying/Monitoring

It is important to note that when a user connects to her account on a remote host using TELNET or FTP, the user's password travels across the Internet unencrypted, or in *plaintext*. Thus, another method for breaking into systems is to monitor connections for IP packets bearing a username and password, and then using them on the system to login normally. If the captured password is to an administrator account, then the job of obtaining privileged access is made much easier. As noted previously, hundreds and possibly thousands of systems across the Internet have been penetrated as a result of monitoring for usernames and passwords.

Electronic mail, as well as the contents of TELNET and FTP sessions, can be monitored and used to learn information about a site and its business transactions. Most users do not encrypt e-mail, yet many assume that e-mail is secure and thus safe for transmitting sensitive information.

The X Window System is an increasingly popular service that is also vulnerable to spying and monitoring. X permits multiple windows to be opened at a workstation, along with display of graphics and multi-media applications (for example, the WWW browser *Mosaic*). Intruders can sometimes open windows on other systems and read keystrokes that could contain passwords or sensitive information.

1.3.4 Ease of Spoofing

As noted in section 1.2.1, the IP address of a host is presumed to be valid and is therefore trusted by TCP and UDP services. A problem is that, using IP source routing, an attacker's host can masquerade as a trusted host or client. Briefly, IP source routing is

forced to scramble to correct the programs before attackers used the vulnerabilities to attack the sites. However, due to the complexity of the sendmail program and networking software in general, three subsequent versions of sendmail were found to still contain significant vulnerabilities [CIAC94a]. The sendmail program is used widely, and sites without firewalls to limit access to sendmail are forced to react quickly whenever problems are found and vulnerabilities revealed.

In the second, a version of a popular and free FTP server was found to contain a Trojan Horse that permitted privileged access to the server. Sites using this FTP server, but not necessarily the contaminated version, were again forced to react very carefully and quickly to this situation [CIAC94c]. Many sites rely on the wealth of free software available on the Internet, especially security-related software that adds capability for logging, access control, and integrity checking that vendors often do not provide as part of the operating system. While the software is often high quality, sites may have little recourse other than to rely on the authors of the software if it is found to have vulnerabilities and other problems.[3]

The third problem has the strongest implications: [CERT94] and [CIAC94b] reported that intruders had broken into potentially *thousands* of systems throughout the Internet, including gateways between major networks, and installed *sniffer* programs to monitor network traffic for usernames and static passwords typed in by users to connect to networked systems. The intruders had used various known techniques for breaking into systems, as well as using passwords that had been "sniffed." One of the implications of this incident is that *static or reusable passwords are obsolete* for protecting access to user accounts. In fact, a user connecting to a remote system across the Internet may be unintentionally placing that system at risk of attack by intruders who could be monitoring the network traffic to the remote system.

The following sections go into more detail on problems with Internet security. [Garf92], [Cur92], [Bel89], [Ches94], and [Farm93] all provide more background and detail; readers are encouraged to consult these references.

1.3.2 Weak Authentication

Incident handling teams estimate that many incidents stem from use of weak, static passwords. Passwords on the Internet can be "cracked" a number of different ways, however the two most common methods are by cracking the encrypted form of the password and by monitoring communications channels for password packets. The UNIX operating system usually stores an encrypted form of passwords in a file that can be read by normal

[3] It should be pointed out that even vendor-supported software has such problems and may be even harder to get fixed in a timely fashion.

whereas clients generally must request unprivileged port numbers from the operating system. Although this rule is not firm and is not required in the TCP/IP protocol specifications, BSD-based systems adhere to it. As an accidental but fortuitous result, firewalls can block or filter access to services by examining the port numbers in TCP or UDP packets and then routing or dropping the packet based on a policy that specifies which services are permitted or denied (this is covered in more detail in Chapter 2).

Not all TCP and UDP servers and clients use ports in as straightforward a fashion as TELNET, but in general the procedure described here is useful in the firewalls context. For example, many personal computer operating systems have no UNIX superuser concept, but still use ports as described (although there is no standard that requires this).

1.3 Security-Related Problems

As stated earlier, the Internet suffers from severe security-related problems. Sites that ignore these problems face some significant risk that they will be attacked by intruders and that they may provide intruders with a staging ground for attacks on other networks. Even sites that do observe good security practices face problems with new vulnerabilities in networking software and the persistence of some intruders.

Some of the problems with Internet security are a result of inherent vulnerabilities in the services (and the protocols that the services implement), while others are a result of host configuration and access controls that are poorly implemented or overly complex to administer. Additionally, the role and importance of system management is often shortchanged in job descriptions, resulting in many administrators being, at best, part-time and poorly prepared. This is further aggravated by the tremendous growth of the Internet and how the Internet is used; businesses and agencies now depend on the Internet (often more than they realize) for communications and research and thus have much more to lose if their sites are attacked. The following sections describe problems on the Internet and factors that contribute to these problems.

1.3.1 Security Incidents on the Internet

As evidence of the above, three problems have occurred within months of each other. In the first, persistent vulnerabilities in the UNIX *sendmail*[2] program were discussed openly on Internet discussion lists. Sites that had not corrected their sendmail programs were

[2] *sendmail* is the mail transport software for most UNIX hosts. It is a very large, complex program that has been found repeatedly to contain vulnerabilities that have permitted intruder access to systems that run sendmail.

How are multiple TELNET connections between two systems identified and coordinated? A TCP or UDP connection is uniquely identified by the following four items present in each message:

- **source IP address** - the address of the system that sent the packet,
- **destination IP address** - the address of the system that receives the packet,
- **source port** - the connection's port at the source system, and
- **destination port** - the connection's port at the destination system.

The port is a software construct that is used by the client or server for sending or receiving messages; a port is identified by a 16-bit number. Server processes are usually associated with a fixed port, e.g., 25 for SMTP or 6000 for X Windows; the port number is "well-known" because it, along with the destination IP address, needs to be used when initiating a connection to a particular host and service. Client processes, on the other hand, request a port number from the operating system when they begin execution; the port number is random although in some cases it is the next available port number.

As an example of how ports are used for sending and receiving messages, consider the TELNET protocol. The TELNET server listens for incoming messages on port 23, and sends outgoing messages to port 23. A TELNET client, on the same or different system, would first request an unused port number from the operating system, and then use this port when sending and receiving messages. It would place this port number, say 3097, in packets destined for the TELNET server so that the server, when responding to the client, could place the client's port number in its TCP packets. The client's host, upon receiving a message, would examine the port and know which TELNET client should receive the message. This is shown conceptually in figure 1.2.

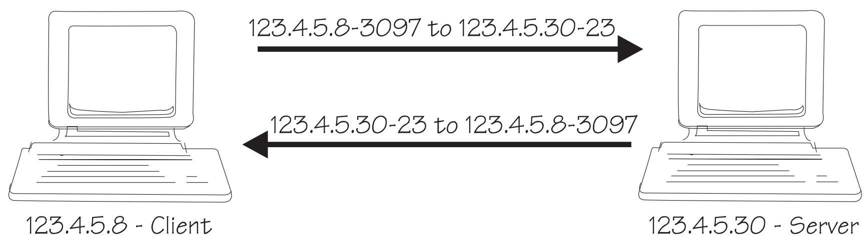

Figure 1.2: TELNET Port, IP Interaction.

There is a somewhat-uniform rule that only *privileged* server processes, i.e., those processes that operate with UNIX superuser privileges, can use port numbers less than 1024 (referred to as *privileged ports*). Servers mostly use ports numbered less than 1024,

1.2.3 UDP

As shown in figure 1.1, UDP interacts with application programs at the same relative layer as TCP. However, there is no error correction or retransmission of misordered or lost packets. UDP is therefore not used for connection-oriented services that need a virtual circuit. It is used for services that are query-response oriented, such as NFS, where the number of messages with regard to the exchange is small compared to TELNET or FTP sessions. Services that use UDP include RPC-based services such as NIS and NFS, NTP (Network Time Protocol), and DNS (DNS also uses TCP).

It is easier to spoof UDP packets than TCP packets, since there is no initial connection setup (handshake) involved (since there is no virtual circuit between the two systems) [Ches94]. Thus, there is a higher risk associated with UDP-based services.

1.2.4 ICMP

ICMP (Internet Control Message Protocol) is at the same relative layer as IP; its purpose is to transmit information needed to control IP traffic. It is used mainly to provide information about routes to destination addresses. ICMP *redirect* messages inform hosts about more accurate routes to other systems, whereas ICMP *unreachable* messages indicate problems with a route. Additionally, ICMP can cause TCP connections to terminate "gracefully" if the route becomes unavailable. *ping* is a commonly-used ICMP-based service.

[Bel89] discusses two problems with ICMP: older versions of UNIX could drop all connections between two hosts even if only one connection was experiencing network problems. Also, ICMP redirect messages can be used to trick routers and hosts acting as routers into using "false" routes; these false routes would aid in directing traffic to an attacker's system instead of a legitimate trusted system. This could in turn lead to an attacker gaining access to systems that normally would not permit connections to the attacker's system or network.

1.2.5 TCP and UDP Port Structure

TCP and UDP services generally have a client-server relationship. For example, a TELNET server process initially sits idle at a system, waiting for an incoming connection. A user then interacts with a TELNET client process, which initiates a connection with the TELNET server. The client writes to the server, the server reads from the client and sends back its response. The client reads the response and reports back to the user. Thus, the connection is bidirectional and can be used for reading and writing.

IP transmits packets that have been received from the TCP or UDP layers to the lower-level layer.

IP packets are *unreliable datagrams* in that IP does nothing to ensure that IP packets are delivered in sequential order or are not damaged by errors. The IP packets contain the address of the host from which the packet was sent, referred to as the *source* address, and the address of the host that is to receive the packet, referred to as the *destination* address.

The higher-level TCP and UDP services generally assume that the source address in a packet is valid when accepting a packet. In other words, the IP address forms the basis of authentication for many services; the services trust that the packet has been sent from a valid host and that host is indeed who it says it is. IP does contain an option known as IP *Source Routing*, which can be used to specify a direct route to a destination and return path back to the origination. The route could involve the use of other routers or hosts that normally would not be used to forward packets to the destination. A source routed IP packet, to some TCP and UDP services, appears to come from the last system in the route as opposed to coming from the true origination. This option exists for testing purposes, however [Bel89] points out that source routing can be used to trick systems into permitting connections from systems that otherwise would not be permitted to connect. Thus, that a number of services trust and rely on the authenticity of the IP source address is problematic and can lead to breakins and intruder activity.

1.2.2 TCP

If the IP packets contain encapsulated TCP packets, the IP software will pass them “up” to the TCP software layer. TCP sequentially orders the packets and performs error correction, and implements *virtual circuits*, or connections between hosts. The TCP packets contain sequence numbers and acknowledgements of received packets so that packets received out of order can be reordered and damaged packets can be retransmitted.

TCP passes its information up to higher-layer applications, e.g., a TELNET client or server. The applications, in turn, pass information back to the TCP layer, which passes information down to the IP layer and device drivers and the physical medium, and back to the receiving host.

Connection oriented services, such as TELNET, FTP, rlogin, X Windows, and SMTP, require a high degree of reliability and therefore use TCP. DNS uses TCP in some cases (for transmitting and receiving domain name service databases), but uses UDP for transmitting information about individual hosts.

and [Bel89] provide more complete descriptions; readers who wish to learn more should consult these references.

Part of the popularity of the TCP/IP protocol suite is due to its ability to be implemented on top of a variety of communications channels and lower-level protocols such as T1 and X.25, Ethernet, and RS-232-controlled serial lines. Most sites use Ethernet connections at local area networks to connect hosts and client systems, and then connect that network via a T1 line to a regional network (i.e., a regional TCP/IP backbone) that connects to other organizational networks and backbones. Sites customarily have one connection to the Internet, but large sites often have two or more connections. Modem speeds are increasing as new communications standards are being approved, thus versions of TCP/IP that operate over the switched telephone network are becoming more popular. Many sites and individuals use PPP (Point-to-Point Protocol) and SLIP (Serial Line IP), to connect networks and workstations to other networks using the switched telephone network.

TCP/IP is more correctly a *suite* of protocols including TCP and IP, UDP (User Datagram Protocol), ICMP (Internet Control Message Protocol), and several others. The TCP/IP protocol suite does not conform exactly to the Open Systems Interconnection's seven layer model, but rather could be pictured as shown in figure 1.1.

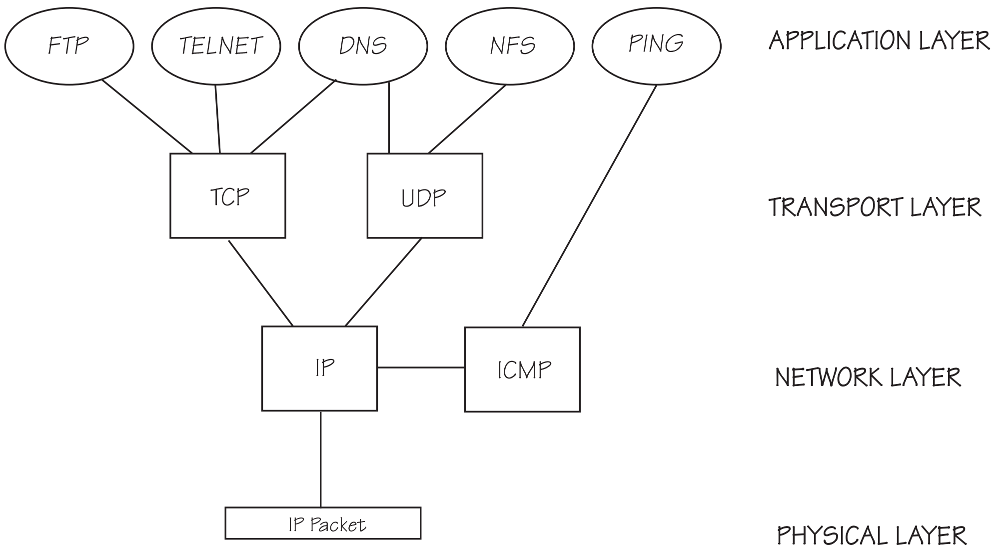

Figure 1.1: Conceptual View of Services and Layers in TCP/IP.

1.2.1 IP

The IP layer receives packets delivered by lower-level layers, e.g., an Ethernet device driver, and passes the packets "up" to the higher-layer TCP or UDP layers. Conversely,

Although TCP/IP can be used equally well in a local area or wide area networking environment, a common use is for file and printer sharing at the local area networking level and for electronic mail and remote terminal access at both the local and the wide area networking levels. Gopher and Mosaic are increasingly popular; both present problems to firewall designers as will be discussed in later sections.

1.1.2 Internet Hosts

Many host systems connected to the Internet run a version of the UNIX operating system. TCP/IP was first implemented in the early 1980's for the version of UNIX written at the University of California at Berkeley known as the Berkeley Software Distribution (BSD). Many modern versions of UNIX derive their networking code directly from the BSD releases, thus UNIX provides a more-or-less standard set of TCP/IP services. This standard of sorts has resulted in many different versions of UNIX suffering from the same vulnerabilities, however it has also provided a common means for implementing firewall strategies such as IP packet filtering. It is important to note that BSD UNIX source code is fairly easy to obtain free from Internet sites, thus many good and bad people have been able to study the code for potential flaws and exploitable vulnerabilities.

Although UNIX is the predominant Internet host operating system, many other types of operating systems and computers are connected to the Internet, including systems running Digital Equipment Corporation's VMS, NeXT, mainframe operating systems, and personal computer operating systems such as for DOS, Microsoft Windows, and for Apple systems. Although personal computer systems often provide only client services, i.e., one can use TELNET to connect *from* but not *to* a personal computer, increasingly powerful personal computers are also beginning to provide, at low cost, the same services as larger hosts. Versions of UNIX for the personal computer, including Linux, FreeBSD, and BSDi, and other operating systems such as Microsoft Windows NT, can provide the same services and applications that were, until recently, found only on larger systems. The ramifications of this are that more people are able to utilize a wider array of TCP/IP services than ever before. While this is good in that the benefits of networking are more available, it has negative consequences in that there is more potential for harm from intruders (as well as uneducated but well-intentioned users who, to some sites, may appear to be intruders).

1.2 Overview of TCP/IP Internals

This section provides a simplified overview of TCP/IP for the purposes of later discussion on Internet-related security problems. [Com91a], [Com91b], [Ford94], [Hunt92],

1.1.1 Common Services

There are a number of services associated with TCP/IP and the Internet. The most commonly used service is electronic mail (*e-mail*), implemented by the Simple Mail Transfer Protocol (SMTP). Also, TELNET (terminal emulation), for remote terminal access, and FTP (file transfer protocol) are used widely. Beyond that, there are a number of services and protocols used for remote printing, remote file and disk sharing, management of distributed databases, and for information services. Following is a brief list of the most common services:

- **SMTP** - Simple Mail Transfer Protocol, used for sending and receiving electronic mail,
- **TELNET** - used for connecting to remote systems connected via the network, uses basic terminal emulation features,
- **FTP** - File Transfer Protocol, used to retrieve or store files on networked systems,
- **DNS** - Domain Name Service, used by TELNET, FTP, and other services for translating host names to IP addresses,
- **information-based services, such as**
  - **gopher** - a menu-oriented information browser and server that can provide a user-friendly interface to other information-based services,
  - **WAIS** - Wide Area Information Service, used for indexing and searching with databases of files, and
  - **WWW/http** - World Wide Web, a superset of FTP, gopher, WAIS, other information services, using the hypertext transfer protocol (http), with *Mosaic* being a popular WWW client,
- **RPC-based services** - Remote Procedure Call services, such as
  - **NFS** - Network File System, allows systems to share directories and disks, causes a remote directory or disk to appear to be local, and
  - **NIS** - Network Information Services, allows multiple systems to share databases, e.g., the password file, to permit centralized management,
- **X Window System** - a graphical windowing system and set of application libraries for use on workstations, and
- **rlogin, rsh, and other "r" services** - employs a concept of mutually trusting hosts, for executing commands on other systems without requiring a password.

1

Introduction to the Internet and Internet Security

While Internet connectivity offers enormous benefits in terms of increased access to information, Internet connectivity is not necessarily a good thing for sites with low levels of security. The Internet suffers from glaring security problems that, if ignored, could have disastrous results for unprepared sites. Inherent problems with TCP/IP services, the complexity of host configuration, vulnerabilities introduced in the software development process, and a variety of other factors have all contributed to making unprepared sites open to intruder activity and related problems.

The following sections present a brief overview of the Internet, TCP/IP, and then explain what some of the Internet security related problems are and what factors have contributed to their seriousness.

1.1 The Internet

The Internet is a world-wide "network of networks" that use the TCP/IP (Transmission Control Protocol/Internet Protocol) protocol suite for communications. The Internet was created initially to help foster communication among government-sponsored researchers. Throughout the 1980's, the Internet grew steadily to include educational institutions, government agencies, commercial organizations, and international organizations. In the 1990's, the Internet has undergone phenomenal growth, with connections increasing faster than any other network ever created (including the telephone network). Many millions of users are now connected to the Internet, with roughly half being business users [Cerf93]. The Internet is being used as the basis for the National Information Infrastructure (NII).

permit more TCP/IP services than they require for their operations and do not attempt to limit access to information about their computers that could prove valuable to intruders, and

- **complexity of configuration** - host security access controls are often complex to configure and monitor; controls that are accidentally misconfigured often result in unauthorized access.

Solutions

Fortunately, there are readily-available solutions that can be used to improve site security. A firewall system is one technique that has proven highly effective for improving the overall level of site security. A firewall system is a collection of systems, routers, and policy placed at a site's central connection to a network. A firewall forces all network connections to pass through the gateway where they can be examined and evaluated, and provides other services such as advanced authentication measures to replace simple passwords. The firewall may then restrict access to or from selected systems, or block certain TCP/IP services, or provide other security features. A well-configured firewall system can act also as an organization's "public-relations vehicle" and can help to present a favorable image of the organization to other Internet users.

A simple network usage policy that can be implemented by a firewall system is to provide access from internal to external systems, but little or no access from external to internal systems. However, a firewall does not negate the need for stronger system security. There are many tools available for system administrators to enhance system security and provide additional logging capability. Such tools can check for strong passwords, log connection information, detect changes in system files, and provide other features that will help administrators detect signs of intruders and break-ins.

Recommendations

NIST recommends that agencies and organizations, prior to connecting to the Internet, develop policy that clearly identifies the Internet services they will be using and how those services will be used. The policy should be clear, concise, and understandable, with a built-in mechanisms for changing the policy. Organizations should strongly consider using firewall systems as part of the implementation of that policy. NIST recommends also that agencies and organizations use advanced authentication measures, i.e., smartcards, or authentication tokens, or other one-time password mechanisms, as an integral part of firewalls for authenticating connections to site systems.

service.

This document uses *application gateways* to refer to some firewall systems as opposed to *bastion hosts.*

As much as possible, this document avoids using terms such as *hacker* and *cracker*, and uses instead the less ambiguous *intruder* and *attacker*.

Background

The Internet is a vital and growing network that is changing the way many organizations and individuals communicate and do business. However, the Internet suffers from significant and widespread security problems. Many agencies and organizations have been attacked or *probed*[1] by intruders, with resultant high losses to productivity and reputation. In some cases, organizations have had to disconnect from the Internet temporarily, and have invested significant resources in correcting problems with system and network configuration. Sites that are unaware of or ignorant of these problems face a significant risk that they will be attacked by network intruders. Even sites that do observe good security practices face problems with new vulnerabilities in networking software and the persistence of some intruders.

A number of factors have contributed to this state of affairs. The fundamental problem may be that the Internet was not designed to be very secure, i.e., open access for the purposes of research was the prime consideration at the time the Internet was implemented. However, the phenomenal success of the Internet in combination with the introduction of different types of users, including unethical users, has aggravated existing security deficiencies to the extent that wide-open Internet sites risk inevitable break-ins and resultant damages. Other factors include the following:

- **vulnerable TCP/IP services** - a number of the TCP/IP services are not secure and can be compromised by knowledgeable intruders; services used in the local area networking environment for improving network management are especially vulnerable,
- **ease of spying and spoofing** - the majority of Internet traffic is unencrypted; e-mail, passwords, and file transfers can be monitored and captured using readily-available software, intruders can then reuse passwords to break into systems,
- **lack of policy** - many sites are configured unintentionally for wide-open Internet access without regard for the potential for abuse from the Internet; many sites

[1] Intruders have been observed to target specific sites for intrusions by methodically scanning host systems for vulnerabilities. Intruders often use automated probes, i.e., software that scans all host systems connected to a site's network. This is sometimes referred to as *probing a site.*

Audience

The intended audience of this publication is technical-level management, i.e., those individuals who may be responsible for implementing or maintaining Internet connections. This document would also be appropriate for other management who wish to learn more about Internet security issues.

Some technical background in computer security and computer network communications is assumed. However, this document is intended to be a starting point; more detailed information about Internet security and firewalls can be found in the references section.

Document Structure

This document begins with an overview of the Internet and common services. It describes Internet-related security problems in detail by examining problems with various TCP/IP services and by examining other factors that have caused the Internet to grow less secure. Chapter 2 discusses firewalls, their benefits as well as their disadvantages, and then the various firewall components, including advanced authentication measures and network access policy. Chapter 3 describes several firewall configurations that illustrate how the firewall components fit together and can be used to implement various policies. Chapter 4 discusses procurement, administrative issues, and other actions sites should take to secure their Internet-connected systems. Appendix A provides pointers to other books and information about firewalls and Internet security. Appendix B contains a collection of frequently asked questions about firewalls that is available on-line (see Appendix B for more information).

Terminology

Internet firewalls are often referred to as *secure Internet gateways* in other literature. This document uses *firewall* to refer to a secure Internet gateway.

A firewall, as defined in this document, includes a number of items such as policy, network arrangement, and technical controls and procedures. This document uses *firewall system* when referring to the hosts or routers that implement the firewall.

This document, when referring to a network protected by a firewall, uses *protected subnet* or *protected LAN* (Local Area Network).

Some people dispute whether TCP/IP protocols should be referred to as protocols or services. It could be argued, for example, that TELNET is a protocol, a service, or a command. Where it makes obvious sense, this document uses *protocol*, otherwise it uses

Preface

The Internet is a world-wide collection of networks that all use a common protocol for communications. Many organizations are in the process of connecting to the Internet to take advantage of Internet services and resources. Businesses and agencies are now using the Internet or considering Internet access for a variety of purposes, including exchanging e-mail, distributing agency information to the public, and conducting research. Many organizations are connecting their existing internal local area networks to the Internet so that local area network workstations can have direct access to Internet services.

Internet connectivity can offer enormous advantages, however security needs to be a major consideration when planning an Internet connection. There are significant security risks associated with the Internet that often are not obvious to new (and existing) users. In particular, intruder activity as well as vulnerabilities that could assist intruder activity are widespread. Intruder activity is difficult to predict and at times can be difficult to discover and correct. Many organizations already have lost productive time and money in dealing with intruder activity; some organizations have had their reputations suffer as a result of intruder activity at their sites being publicized.

This publication focuses on security considerations for organizations considering Internet connections as well as for organizations already connected to the Internet. In particular, this document focuses on Internet *firewalls* as one of the mechanisms and methods used for protecting sites against Internet-borne threats. This document recommends that organizations use firewall technology and other related tools to filter connections and limit access. This document is an expansion of the issues and guidance contained in NIST CSL Bulletin, *Connecting to the Internet: Security Considerations* [NIST93].

Purpose

The purpose of this document is to provide a basis of understanding of how firewalls work and the steps necessary for implementing firewalls. Users can then use this document to assist in planning or purchasing a firewall. This document does not explain how to build a firewall; references are provided for more detailed information.

List of Figures

Contents

Acknowledgments

The National Institute of Standards and Technology would like to thank the following individuals who reviewed drafts of this document and advised on document structure and content: David Curry of Purdue University, Uwe Ellermann of the DFN-CERT in Germany, and Stephen Weeber of the Department of Energy's Computer Incident Advisory Capability (CIAC).

Abstract

This document provides an overview of the Internet and security-related problems. It then provides an overview of firewall components and the general reasoning behind firewall usage. Several types of network access policies are described, as well as technical implementations of those policies. Lastly, the document contains pointers and references for more detailed information.

The document is designed to assist users in understanding the nature of Internet-related security problems and what types of firewalls will solve or alleviate specific problems. Users can then use this document to assist in purchasing or planning a firewall.

NIST Special Publication 800-10
U.S. DEPARTMENT OF COMMERCE
National Institute of Standards and Technology

Keeping Your Site Comfortably Secure: An Introduction to Internet Firewalls

John P. Wack
Lisa J. Carnahan