NEAR EAST UNIVERSITY

Faculty of Engineering

Department of Computer Engineering

FIREWALLS AND NETWORK SECURITY

Graduation Project
COM-400

Student: Devrim Gücal (970210)

Supervisor: Prof.Dr. Fakhreddin Mamedov

Lefkoşa-2003

5.1.1. How packet filtering works 28
5.1.2. What services to filter? 29
5.1.3. A few rules for filtering by service 30
5.1.4. Protocol specific issues for filtering Telnet traffic 31
5.1.5. IPRoute packet filtering 33
5.2.Proxy systems 34
5.2.1. Bastion host features 35
5.2.2. How a proxy system works 35
5.2.3. Custom user procedures vs. custom client 36
5.2.4. Circuit-level gateway 37
5.2.5. SOCKS 38
5.3.Stateful multi-layer inspection 39
6. Benefits and limitations of firewalls 40
6.1.Benefits of firewalls 40
6.1.1. Benefits of packet filtering routers 40
6.1.2. Benefits of proxy systems 41
6.2.Limitations of firewalls 41
6.2.1. Limitations of packet filtering routers 41
6.2.2. Limitations of proxy systems 42
7.Firewall architecture 43
7.1.Introduction 43
7.2.Dual-homed host 43
7.3.Screened host 44
7.4.Screened subnet
PART IV. APPENDICES
APPENDIX A Example IPRoute configuration 77
APPENDIX B Test sessions to/from the Guardian Firewall 82
APPENDIX C Test sessions to/from the AltaVista Firewall 85
APPENDIX D Proposal for Master Project 90
Acronyms 98
Reference 99

ACKNOWLEDGEMENTS

Respectfil thanks to my parents who have contributrd so much effort to bring me up,to my teachers for their help in academic life and Mr. Fakhreddin Mamedov,who supported me with his knowledge in this project.

ABSTRACT

This paper is a proposal for graduation project in which network security and firewalls will be analyzed as a most effective way for addressing network security problems.

The proposal will include a discussion of the motives for research on firewalls as well as an overview of some firewall products. The project will be implementation oriented and will assist in understanding the nature of network security problems and what types of firewalls will solve or alleviate specific problems.

ACRONYMS

ARP – Address Resolution Protocol

BSD – Berkeley Software Distribution

DES – Data Encryption Standard

DNS – Domain Name Service

DSS – Digital Signature Standard

FTP – File Transfer Protocol

HTTP – HyperText Transfer Protocol

ICMP – Internet Control Message Protocol

IRC – Internet Relay Chat

ISN – Initial Sequence Number

LAN – Local Area Network

MAC – Message Authentication Code

MBONE – Multicast Backbone

NAT – Network Address Translator

NFS – Network File System

NIC – Network Interface Card

NIC – Network Information Center

NIS/YP- Network Information Service/Yellow Pages

NNTP – Network News Transfer Protocol

NTP – Network Time Protocol

NVT – Network Virtual Terminal

OSI – Open System Interconnection

RARP – Reverse Address resolution Protocol

RFC – Request for Comments

RPC – Remote Procedure Call

RSA – Rivest, Shamir, Adleman

SAH – Secure Hashing Algorithm

SMLI – Stateful Multi-Layer Inspection

SMTP – Simple Mail Transfer Protocol

SNMP – Simple Network Management Protocol

TCP/IP – Transmission Control Protocol/Internet Protocol

TFTP – Trivial File Transfer Protocol

UDP – User Datagram Protocol

WAIS – Wide Area Information Service

WAN – Wide Area Network

WWW – World Wide Web

1. THE INTERNET

1.1. Introduction

The Internet is one of the most important developments in the history of information systems. The Internet is not one network, but rather a worldwide collection of Networks that all use a common protocol for communications. Use of a common protocol among incompatible network technologies opened the possibilities of shared resources in the computing industry, and has given rise to a whole new level of connectivity in the workplace. The Internet has become a common ground for information exchange.

Although many protocols have been adapted for use in an internet, one suite known as TCP/IP (Transmission Control Protocol / Internet Protocol), stands out as the most widely used for interconnection of many disparate physical networks. TCP/IP is the glue that holds the Internet together and makes universal service possible. TCP/IP technology has made possible a global Internet that includes over 10,000 different networks in more than 100 different countries.

The Internet started out as U.S. Department of Defense network that connected research scientists and academics around the world. Originally, commercial traffic was forbidden on the Internet because the key portions of the network were funded by the U.S. government. Today the Internet is no longer maintained by the government, but rather by a private industry consortium, and everyone can join the Internet by paying a registration fee and agreeing to maintain certain communication standards. The benefits of connecting to the Internet range from lower communication cost and greatly improved communication to the vast variety of the Internet services and resources.

The Internet organization is based on a hierarchy at whose root lie providers. The Internet's providers connect their networks to form the worldwide backbone for the Internet. Individual provider networks may be limited to small geographic regions or they may span entire continents.

1.2. Internet services

There are a number of services associated with the Internet that users want to access. The most popular and commonly used Internet application services include electronic mail, file transfer, remote terminal access, and World Wide Web access. Beyond that, there are a number of services used for remote printing, transferring news,

conferencing, management of distributed databases and information services. Following is a brief summary of the major Internet services that users may be interested in using.

- Electronic mail is implemented using Simple Mail Transfer Protocol (SMTP) which is Internet standard protocol for sending and receiving electronic mail.
- File transfer is the method designed for transferring files on request. File Transfer. Protocol (FTP) is the Internet standard protocol for this purpose.
- Remote terminal access is used for connecting to remote systems connected via the network, as if they were directly attached. TELNET is the standard for remote terminal access on the Internet. There are other programs that are used for remote terminal access and remote execution of programs such as rlogin, rsh, and other "r" commands (rcp, rdump, rrestore, rdist).
- Name service is what translates between the host names that people use and the numerical IP addresses that machines use. Domain Name Service (DNS) is not a user level service, but it is used by TELNET, SMTP, FTP and every other service that a user needs.
- Network News Transfer Protocol (NNTP) is used to transfer news across the Internet.
- Information services such as
    1. Gopher which is a menu-oriented tool that helps users find information on the Internet.

    - WAIS that stands for Wide Area Information Service and is used for indexing and searching with databases of files.

    - Archie which is an Internet service that searches indexes of anonymous FTP servers for file and directory names.

    - World Wide Web (WWW) is based in part on existing services, and in part on a new protocol, HyperText Transfer Protocol (HTTP). Web servers are accessed by Mosaic, Netscape Navigator and other popular web browsers.

    - Finger service which looks up information about a user who has an account on the machine being queried

    - Who is service which is similar to finger, but it obtains publicly available information about hosts, networks, domains and their administrators.

- Real time conferencing services

  - Talk is the oldest real-time conferencing system used on the Internet which allows two people to hold a conversation.

  - Internet Relay Chat (IRC) involves lots of people talking to each other.

  - New set of services provided over Multicast Backbone (MBONE), which is focused on expending real-time conference services beyond text-based services, like talk and IRC, to include audio, video, and electronic whiteboard.

- Remote Procedure Call (RPC)-based services.

  - Network File System (NFS) which allows systems to access files across the network on a remote system, as if the files were on directly attached disks.

  - Network Information Service / Yellow Pages (NIS/YP) is designed to provide distributed access to centralized administrative information shared by machines as a site.

- Network Management Services are services that most users don't use directly, but rather, they allow network managers to debug problems, control routing, and find computers that violate protocol standards. The most widely used is the Simple Network Management Protocol (SNMP) which is designed to make it easy to centrally manage network equipment.

- Time service is implemented using Network Time Protocol (NTP). NTP is an Internet service that sets the clock on one's system with great precision.

- Printing service provides remote printing options. Bot the system V printing system and the Berkeley Software Distribution (BSD) printing system allow a computer to print to a printer that is physically connected to a different computer.

Because these services form an integral part of TCP/IP, we will defer more detailed description of the most popular to a later section (2.5) where the application layer of TCP/IP architecture is discussed.

1.3. Internet hosts

A host is a computer system that runs applications, is connected to an internet, and has one or more users. A host that supports TCP/IP can act as the endpoint of a communication. Because Personal Computers (PCs), workstations, minicomputers, and

mainframes satisfy the above definition, and all can run TCP/IP, they all can be a host. Different literature refers to the host as a station, computer, or computer system.

Many hosts connected to the Internet run a version of the UNIX operating system. Although UNIX is the predominant Internet host operating system, many other types of operating systems and computers are connected to the Internet. This includes, for example, systems running VMS, other mainframe operating systems and personal computer operating systems such as DOS and Windows. Even more, some versions of UNIX for personal computers and other operating systems such as Microsoft Windows NT can provide, to the increasingly powerful PC, the same services and applications that were recently found only on larger systems. Internet hosts have not only a difference in operating systems they run, but also a host's CPU can be slow or fast, and the size of memory that a different host can have can be different. Fortunately, in spite of all these differences, the TCP/IP protocol allows for any pair of hosts on the Internet to communicate.

2. TCP/IP OVERVIEW

2.1. Introduction

Although many protocols have been adapted for use in an internetl, the Transmission Control Protocol / Internet Protocol (TCP/IP) suite of data communications protocols is currently the most widely used set of protocols for internetwork communication. The name TCP/IP is derived from two of the protocols that belong to it: the Transmission Control Protocol and the Internet Protocol.

TCP/IP evolved from work done in the network research community, in particular the late '60s and early '70s work on packet switching that led to development of ARPANET (ARPA is an acronym for the Advanced Research Projects Agency). The ARPANET was at the beginning a research network sponsored by the DoD (U.S. Department of Defense), but eventually connected hundreds of universities, organizations, and government installations. ARPANET was a packet switched network, but it was a single network and it used protocols not intended for internetworking. In the mid '70s network researchers realized that various LAN technologies (e.g. Ethernet) were starting to be widely deployed, as well as satellite and radio networks. The existing protocols had trouble with internetworking, so new a reference architecture with ability to connect multiple networks together in a seamless way was needed. TCP/IP, a true internetworking protocol suite, is the product of these changes in the networking environment.

Widespread deployment of TCP/IP occurred within the ARPANAET community in the early '80s. By 1983 the name Internet came into use as the official name of the community of interconnected networks using TCP/IP. The Internet demonstrates the viability of the TCP/IP technology and shows how it can accommodate a wide variety of underlying network technologies.

2.2. TCP/IP protocol architecture

Like any modern communication protocol, TCP/IP is a layered protocol. It is also called the Internet layering model or the Internet reference model. This model resembles, but is not the same as the Open System Interconnection (OSI) seven-layer model. Generally it has been composed of fewer layers than the OSI model, and most descriptions of TCP/IP define three to five functional layers in the protocol architecture. Each layer on one machine carries on a conversation with a corresponding layer on another machine. The

rules and conventions used in this conversation are known as the protocol of each separate layer. The five layer model is illustrated in Figure 2.1 below.

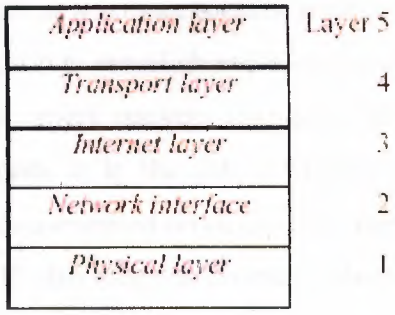


Figure 2.1. The five layers of the TCP/IP protocol architecture

Not only the number of layers differ from the OSI model, but also the name, the contents, and the function of each layer differ. However, in both networks, the purpose of each layer is to offer certain services to the higher layer, shielding those layers from the details of how the offered services are actually implemented. Thus each layer has its own independent data structure and its own terminology to describe that structure.

Data is passed down the stack when it is being sent to the network and up the sack when it is being received from the network. Each layer in the stack adds control information (header), placed in the front of the data to be transmitted, to ensure proper delivery. Each layer treats all of the information it receives from the layer above as data and places its own control information in front of it. When data is received, each layer strips off its header before passing the data on to the layer above.

2.3. Internet layer

2.3.1. Internet Protocol

The Internet Protocol (IP) is the heart of the TCP/IP suite and the most important protocol in the Internet layer. IP provides essential transmission services on which TCP/IP networks are built and all the protocols above and below it depend on its services. IP provides many additional transmission services such as: enriched addressing, defining of packet format, performing fragmentation and reassembly in order to overcome any limitations placed by the data link upon the size of a frame.

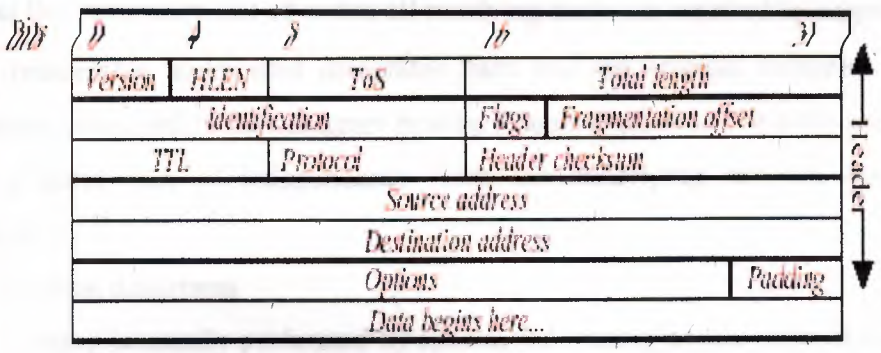
It is also possible, using Internet layer services, to create internetworks of independent LANs and send packets from a node on one LAN to a node on another. This requires routers which forward packets based upon their destination IP address.

IP is a connectionless protocol, which means that IP does not exchange control information to establish end-to-end connection before transmitting data. Its job is to permit hosts to inject packets into any network and have them travel independently to the destination. It is the job of higher layers to establish the connection if they require connection-oriented service and to rearrange the packets if they arrive in a different order.

IP also relies on protocols above it to provide error detection and error recovery.

- IP packet format

IP defines a specific packet format and at this layer of the protocol stack they are called datagrams. An IP datagram consists of header followed by arbitrary data, as illustrated in Figure 2.2.



Notes:

HLEN Header length

ToS Type of service

TTL Time to live

Figure 2.2. IP datagram format

An IP header is five or six 4-byte words long and is padded if necessary. The header contains all the information needed to deliver the packet. Thus, a packet can be routed on an internet without reference to any other packet. This has some implications for the

transport layer because IP does not guarantee delivery or the order of delivery. It is up to the transport layer to perform these tasks.

- Fragmentation and reassembly of datagrams

An IP datagram in transit may traverse different networks whose maximum packet size is smaller than the size of the datagram. To handle this, IP provides fragmentation and reassembly mechanisms. If the datagram received from one network is longer than what the other network can accommodate as a single packet, IP must divide the datagram into smaller fragments for transmission. This process is called fragmentation, and smaller pieces of a datagram are called datagram fragments.

The format of each fragment is the same as the format of any normal datagram. Several fields in the datagram header contain information that identifies each datagram fragment.

Because IP datagrams may be routed independently and fragmented datagrams may arrive at the destination out-of-order, all receiving hosts are required to support reassembly. IP will reassemble fragmented datagrams back into the original datagram based on the information contained in the datagram header. Fragmentation can be quite expensive, but it allows a great deal of independence from the underlying network layer protocol's limitations.

- Routing datagrams

Routing is usually performed by specialized routing nodes, referred to as IP routers because they use IP to route packets between networks. When a router receives an IP packet, it examines the destination IP address in the IP packet header. If the address is one of the locally attached networks, the router just forwards the packet to the host on the local network.

If the destination network number is not a locally attached network, the IP router consults a routing table to determine where to send the packet. This, of course, requires consistent routing tables to be maintained on all IP routers in the internet. This can be done statically and dynamically. Static routes are manually created routing table entries, while dynamic routing uses a routing update protocol to keep all routers aware of the topological changes or routing node failures. Routing issues are very complex and particularly in a

large internetwork like the Internet. Routing authority itself can be distributed across the entire Internet.

2.3.2. Other protocol at the IP layer

There are three other important protocols available at the internet layer: Internet Control Message Protocol (ICMP), Address Resolution Protocol (ARP), and Reverse Address Resolution (RARP)

- ICMP

Packet recipients use ICMP to inform the sender about some errors encountered, flow control problems, detection of unreachable destination and other perceived problems. This may be perceived by the destination host or an intermediate router. ICMP is a functional part of the IP layer, but it uses the IP datagram delivery facility to send its messages. An ICMP message travels in the data area of an IP datagram, and datagrams carrying ICMP messages are routed exactly like datagrams carrying information for users; there is no additional reliability or priority.

Although each ICMP message has its own format, all start with the same three fields: a type field - that identifies the message; a code field - that sometimes provides more specific description of the error; and a checksum field. The format of the rest of the message is determined by the type field. Technically ICMP is an error reporting mechanism. The gateway uses ICMP to inform the original source that a problem has occurred. ICMP includes echo request/reply messages, destination unreachable messages, source quench messages - that control the flow, and redirect messages - that request a host to change its routing tables. Echo request/reply is one of the most frequently used debugging tools to determine whether destination can be reached. ICMP also can inform the sender of preferred routes or of network congestion.

- ARP

The Internet behaves like a virtual network, using only those addresses assigned by the IP addressing scheme when sending and receiving data. When a host or a router needs to transmit a frame across a physical network, it should map an IP address to the correct physical or hardware address. The Address Resolution Protocol (ARP) provides a method for dynamically translating between IP addresses and physical addresses.

There are three groups of address resolution algorithms that depend on the type of physical address scheme used. In the first mechanism, hardware addresses may be obtained by looking at a table that contains address translation information. The second mechanism, called closed-form computation, establishes direct mapping by having the machine's physical address encoded in its IP address. In the third approach, mapping is performed dynamically, i.e. a computer that needs to resolve an address sends a message across a network and receives a reply. Table look up is usually used to map WAN addresses, closed-form computation method is used on the networks with configurable hardware addresses, and message exchange is used on LANs with static addressing. To reduce network traffic and make ARP efficient, each machine saves temporarily IP-to physical address bindings in its ARP table.

When a host wants to start communication with another machine, it looks for that machines IP address in its ARP table of bindings in RAM memory first. If there is no entry for that IP address, the host broadcasts an ARP request containing the destination IP address. The target machine that recognize its IP address responds to the request by sending replies that contain its own hardware interface address.

- RARP

A variant of ARP called reverse ARP was designed to help a node to find out its own IP address before it could communicate using TCP/IP. Because a machine's IP address is usually kept on its secondary storage RARP, was intended for use by diskless workstations and other devices that need to get configuration information from a network server.

A station using the reverse ARP protocol, broadcasts a query to all machines on the local network stating its physical address, and requesting its IP address. One or more servers that are configured with a table of physical addresses and watching incoming IP addresses, reply to the sender.

2.4. Transport layer

The layer above the internet layer in the TCP/IP model is called the transport layer. The transport layer is designed to provide reliable and efficient end-to-end subnetindependent connection and transaction services. The transport layer has two

principal protocols: Transmission Control Protocol (TCP) and User Datagram Protocol (UDP).

Both protocols deliver data between the application layer and the internet layer.

Application programmers can choose whichever service is more appropriate for their specific applications.

2.4.1. TCP

TCP is designed to operate over a wide variety of networks and to provide reliable, connection-oriented transmission of user data. TCP allows a byte stream originating on one machine to be delivered without error on any other machine in the Internet. TCP is also responsible for passing data to and from the correct application. The application for which data are sent is identified by a 16-bit number called the port number. The source port and destination port are contained in the segment header.

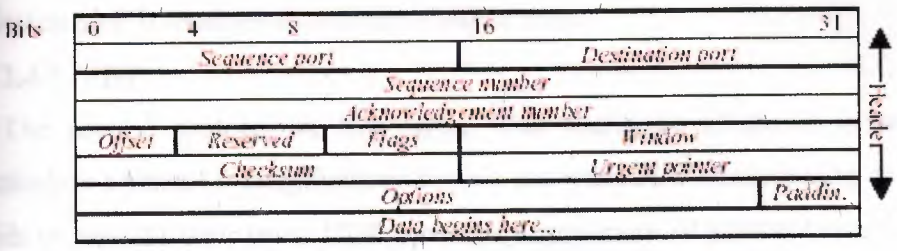


Figure 2.3. TCP segment format

TCP provides reliability by employing a Positive Acknowledgement with Retransmission (PAR) mechanism to recover from the loss of data by the lower layers. A system using PAR allows a sending host's TCP to retransmit data at timed intervals, unless a positive acknowledgement is returned. The unit of data exchanged between cooperating TCP modules is called a segment (see Figure 2.3.). Each segment contains a checksum that detects data segments damaged in transit. If the data segment is received damaged, the receiver discards it without acknowledgement. PAR, therefore, treats damaged segments the same as lost segments and compensates for their loss. The sequence numbers used by TCP extend the PAR mechanism by allowing a single acknowledgement to cover all previously received data.

TCP builds a virtual circuit on top of the unreliable packet-oriented service of IP, by initializing and synchronizing the connection information between the two communicating

hosts. Control information, called a handshake, is exchanged between two endpoints to establish a dialogue before data is transmitted. The procedure used in TCP is called a three-way handshake because the two communicating hosts synchronize sequence numbers by exchanging three segments. The three-way handshake works on the basis that both machines, when attempting to open a communication channel, transmit sequence numbers (seq) and acknowledgement numbers (ack). This procedure reduces the possibility that a delayed packet will appear as a valid packet within the current connection.

TCP also incorporates a flow control algorithm that makes efficient use of available network bandwidth. This algorithm is based on a window which defines a contiguous range of acceptable sequence numbered data. The window indicates to the sender that it can continue sending segments as long as the total number of bytes that it sends is smaller than the window of bytes that the receiver can accept. A zero window tells the sender to stop transmission until it receives a non-zero window value.

2.4.2. UDP

The second protocol in this layer, User Datagram Protocol, is an unreliable, connectionless protocol for applications that do not want TCP's sequencing or flow control and wish to provide their own. UDP provides a minimum of protocol overhead to allow applications to exchange messages over the network. UDP is an unreliable protocol, which means that there are no techniques in the protocol for verifying that the data reached the other end of the network. The only type of reliability is that UDP performs a simple checksum of each message.

Like in TCP, UDP is responsible for delivering data to and from the application layer. It also uses 16-bit source port and destination port numbers in the message header (see Figure 2.4.), to deliver data to the correct application process. The UDP protocol is used in situations where the amount of data being transmitted is small. In such cases the overhead of creating connections and ensuring reliable delivery.

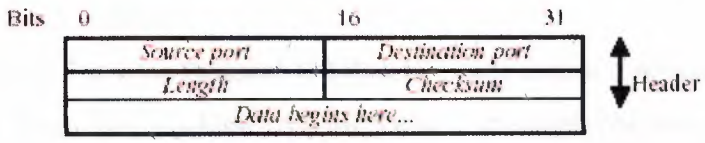


Figure 2.4. UDP datagram

May be greater than the work of retransmitting the entire data if it is received incorrectly. Thus UDP is widely used for one-shot, client-server type request-reply queries and applications in which prompt delivery is more important than accurate delivery, such as transmitting speech or video.

2.5. Application layer

Layer five of the TCP/IP protocol architecture is the application layer. The application layer consists of a number of applications and processes that use the network to deliver data. All of these are built on top of transport layer protocols, either TCP or UDP. In chapter 1.2 we already mentioned the number of user services and application protocols that support them, but the most widely known and implemented application protocols are Telnet, FTP, SMTP, and DNS.

2.5.1. Telnet

Telnet is one of the oldest of the TCP/IP protocols and was adapted from a protocol that had the same name and that was used in the original ARPANET. In comparison with some other remote terminal protocols, Telnet is not as sophisticated, but it is widely available, and it is standard on the Internet.

Telnet allows a user from any Internet-connected site to log into a server at another site. A user establishes a TCP connection, which allow him to use a remote system as if it were directly attached. Telnet relies primarily on TCP to establish a connection with a remote machine that allows use of a remote system as if it were directly attached. Because of differences between computers and operating systems, Telnet defined a Network Virtual Terminal (NVT) as one which will provide a standard interface to remote systems. NVT actually maps the differences between various local terminals to a common convention . Another important service that Telnet offers is options negotiation between the client and server. It provides a wide range of options such as transmit 8-bits data instead of , allows one side to echo data it receives, operates in half- or full-duplex mode, etc.

2.5.2. FTP

File Transfer Protocol (FTP) lets a user access a remote machine and transfer files to and from that machine. As for Telnet, standard file transfer protocol existed in the

ARPANET, which eventually developed into FTP. Currently, FTP is probably among the most frequently used TCP/IP applications.

There are two types of FTP access: user FTP and anonymous FTP. User FTP requires an account on the server and users have to identify themselves by sending a login name and password to the server before requesting any file transfer. After that, the users can Access any files they are allowed to access as if they were logged in. Anonymous FTP Access means that the user does not need an account or password. Anonymous FTP is used by many sites to provide unrestricted access to specific files to the public. Anonymous FTP is the most common mechanism on the Internet to allow remote access to publicly available information and other files.

FTP uses two separate TCP connections: one to carry commands between client and server which is usually called the control channel, and the other to carry any actual files usually called the data channel. The control channel persists throughout the overall session, while data channels can be established dynamically for each new file transfer. To open the control channel connection to the server, the client uses a locally assigned port for itself, but contacts the server at well-known port 21. The data channel normally uses port 20.

Besides FTP there is a simplified version of it, called Trivial File Transfer Protocol (TFTP). TFTP is more restrictive and consequently TFTP software is much smaller than FTP. This small size enables TFTP to be built into hardware, so that diskless machines can use it to transfer information.

2.5.3. SMTP

Electronic mail is probably the most popular and the most fundamental network service. On the Internet, electronic mail exchange between client and server is handled with a standard transfer protocol known as Simple Mail Transfer Protocol (SMTP). Communication between client and server consists of readable text. That means that although SMTP defines that messages sent begin with a command format, usually a 3-digit number that the program uses, they are followed by text that humans can easily read to understand interaction.

To provide for interoperability across the widest range of computer systems and networks, this standard transfer protocol is divided into two sets. One set specifies the exact

format for mail messages, while the other specifies how the underlying mail delivery system passes messages across a link from one machine to another.

Separation of the standard in two parts is extremely useful for providing connection among standard TCP/IP mail systems and other vendors' mail systems, or between TCP/IP networks and networks that do not support this protocol. In such cases it is possible to place a mail gateway which will accept mail messages from the private network and forward them to the Internet, using the same message format for both.

SMTP is the forwarding system. Whenever the user sends or receives a mail message, the system places a copy in its storage (spool) area: outgoing spool area for outgoing mail and mailboxes for incoming mail. But before an incoming or outgoing mail message is placed into one of a spool areas, it passes through the mail forwarder. Delivery address is first recorded into the proper form, and then is examined to decide whether to deliver the mail locally i.e. to place the message in the incoming mailbox, or to forward it to some other machine, i.e. to place the message in the outgoing spool area.

2.5.4. DNS

Domain Name Service relies on simple protocol, which allow clients to send questions to the server, and servers to respond with answers. Users generally do not use this service directly, but it underlies Telnet, FTP, SMTP and every other service, by mapping the Internet host names to their corresponding IP addresses and vice versa. Thus this service allows users to identify systems with simple human-readable names.

But DNS provides more than a translation service. It also defines a hierarchical name space that allows distribution of naming authority and organizes the name servers that implement the DNS protocol. Consequently, DNS has two independent aspects. To efficiently map names to addresses DNS first, specifies the name syntax and rules for delegating authority over names, and second, it includes a set of servers operating at multiple sites.

The hierarchical naming scheme known as domain names consists of a sequence of subnames separated by a delimiter character, the period. The Internet domain name hierarchy is a tree-like structure, at the top of which are seven top-level domains. Figure 2.5 lists those domains and shows their meaning. The Internet also supports, as top-level domain names, two-letter country codes. Thus, the top-level names permit two completely

different naming hierarchies: geographic and organizational. Domain names are written with the local label first and the top domain last.

The DNS also organizes the name servers in a tree structure that corresponds to the naming hierarchy. At the top of this tree is the root server that has responsibility to supply name-to-address translation for the entire Internet. Given the name to resolve, the root can choose the correct name server, each of which translate names for one top-level domain, and thus delegates some of the responsibility. At each of the next levels, name servers can resolve subdomains under its domain. The hierarchy of names ensures their uniqueness of names, and the hierarchy of servers prevents every server from having to know every name.

DNS can use either UDP or TCP to communicate. Usually when some query arrives, the local name server responds using the same transport service as the request. Both queries and responses use the same message format. This format allows a client to ask multiple questions in a single message. Each question consists of a domain name for which the client seeks an IP address followed by the query type and query class.

Domain Name	*Meaning*
COM	*Commercial organization*
EDU	*Educational institution*
GOV	*Government institution*
MIL	*Military groups*
NET	*Network providers*
INT	*International organizations*
ORG	Other organizations

Figure 2.5. The top-level Internet domains and their meaning

2.6. The IP addresses

To deliver data between two Internet hosts it is necessary to have some kinds of addresses that contain sufficient information to uniquely identify every host on the Internet. TCP/IP uses a scheme in which each host is assigned a 32-bit address called its Internet address or IP address. IP addresses are usually written as four decimal numbers separated by dots, where each integer gives the value of one byte of the IP address.

An IP address contains a network part and a host part. The number of bits used to identify these parts depends on the class of address. There are three main address classes: class A, that devote first byte for network and the next three bytes for host address; class B which allocates first two bytes to identify the network and the last two bytes to indicate the

host; and finally, class C which allocate the first three bytes for network address and the last byte for host number. Not all of these addresses are available for use. Some of them, that include a combination of 0's and 1's, are reserved for special uses such as limited broadcast, loopback for testing purposes, etc. To insure that the network portion of an Internet address is unique all Internet addresses are assigned by a central authority, the Network Information Center (NIC).

Unfortunately, this address format with fixed size of 32 bits on which IPv4 relies has placed a limit on the Internet's growth. IPv6 overcomes this limitation by increasing the size of network addresses. IPv6 are 128 bits long, and it is believed that this size will accommodate network addresses for even the most pessimistic estimates of the Internet growth.

3. ELEMENTS OF NETWORK SECURITY

3.1. Why we need secure networks

In recent years organizations have become increasingly dependent on the Internet for communications and research. Regardless of the organization type, users on private networks are demanding access to Internet services such as Internet mail, Telnet and File Transfer Protocol. In addition, because of Internet's powerful and easy available medium, many organizations use it for business transactions. The Internet has also opened possibilities of efficient use and availability of shared resources across a multi-platform computing environment. The recent explosion of the World Wide Web is responsible, in large part, for further tremendous growth of the Internet and even bigger needs for accessing it.

With the spread of Internet protocols and applications, there has been a growth in their abuse as well. Dependence of an organization on the Internet has changed the potential vulnerability of the organization's assets, and security has become one of the primary concerns when an organization connects its private network to the Internet. Connection to the Internet exposes an organization's private data and networking infrastructure to Internet intruders. Many organizations have some of their most important data, such as their financial records, research results, design of new products, etc., on their computers which are attractive for attackers who are out there on the Internet.

A wide variety of threats face computer systems and the information they process which can result in significant financial and information losses. Threats vary considerably – from threats to data integrity resulting from unintentional errors and omissions, to threats to system availability from malicious hackers attempting to crash a system. Knowledge of the types of threats and vulnerabilities aids in the selection of the most cost-effective security measures.

Security is concerned with making sure that "nosy" people cannot break into the organization's private network, read or steal confidential data or worse yet, modify it in order to sabotage that organization. It also deals with other types of attacks. Examples include service interruption, interception of sensitive e-mail or data transmitted, use of computer's resources and so on.

Most network based computer security crimes are unreported. Companies do not want to reveal that their computer systems and data have been compromised. Even if a company's data isn't damaged and attackers didn't actually do anything to computer infrastructure, there are serious consequences of breaches. The most serious would be shaking people's confidence in that organization.

3.1.1. Security problems

The Internet suffers from severe security-related problems. Some of the problems are a result of inherent vulnerabilities in the TCP/IP services, and the protocols that the services implement, while others are a result of the complexity of host configuration and vulnerabilities introduced in the software development process. These and a variety of other factors have all contributed to making unprepared sites open to the Internet attackers . The Internet attacks range from simple probing to extremely sophisticated forms of information theft.

The TCP/IP protocol suite, which is very widely used today, has a number of serious security flaws. Some of these flaws exist because hosts rely on IP source address for authentication, while others exist because network control mechanisms have minimal or non-existent authentication. Unfortunately some individuals have taken advantage of potential weaknesses in the TCP/IP protocol suite and have launched a variety of attacks based on these flaws. Some of these attacks are:

- TCP Initial Sequence Number (ISN) guessing: When a virtual circuit is created in a TCP environment, the two hosts need to synchronize the Initial Sequence Number (ISN). However, there is a way for an intruder to predict the ISN and construct a TCP packet sequence without ever receiving any responses from the server. This allowed an intruder to spoof a trusted host on a local network. Reply messages are received by the real host, which will attempt to reset the connection. Prediction of the random ISN is possible because in Berkeley systems, the ISN variable is incremented by a constant amount once per second, and by half that amount each time a connection is initiated. Thus, if one initiates a legitimate connection and observes the ISN used, one can calculate, with a high degree of confidence, ISN used on the next connection attempt.

Some other people can be purely curious. They will break in just to learn about an organization's computer system and data, or because they like the challenge of testing their skills and knowledge. Breaking into something well known and well defended is usually worth more to this kind of intruder. But also there are professional hackers, sometimes called crackers, whose breeches are much more serious and dangerous. They break into corporate or government computers for specific purposes such as espionage,fraud, and theft. One study of a particular Internet site found that hackers attempted to break in once at least every other day .

Obviously, most security problems are intentionally caused by malicious people trying to gain some benefit or harm someone. Making a network secure involves a lot of effort. Developing a secure network means developing mechanisms that reduce or eliminate the treats to network security. The right approach to network security should include building firewalls to protect internal systems and networks, using strong authentication methods, and using encryption to protect particularly sensitive data as it transits the network.

3.2. Security policy

Before implementing any security tools, software, or hardware, an organization must have some security plan. A site security plan could be developed only after an organization has determined what it needs to protect and the level of protection that it needs. Request for Comments (RFC) 1244 is a site security handbook, that provides guidance to site administrators on how to deal with security issues on the Internet .

A security policy is an overall scheme needed to prevent unauthorized users from accessing resources on the private network, and to protect against unauthorized export of private information. A security policy must be part of an overall organization security scheme; that is, it must obey existing policies, regulations and laws that the organization is subjected to.

A site security policy is needed to establish how both internal and external users interact with a company's computer network, how the computer architecture topology within an organization will be implemented, and where computer equipment will be located. One of the goals of a security policy should be to define procedures to prevent and

respond to security incidents. It is very important that once a security policy is developed and in place, it must be obeyed by everyone from that organization.

3.2.1. Stances of security policy

There are two opposed stances that a security policy can take to describe the fundamental security philosophy of the organization .

- That which is not specifically permitted is prohibited. This stance assumes that the security policy should start by denying all access to all network resources, and then each desired service should be implemented on a specific basis. This is the beter approach.
- That which is not specifically prohibited is permitted. This stance assumes that the security policy should permit access to all network resources, and then each potentially dangerous service should be prohibited on a case-by-case basis. This approach provides for more services available to the users, but it makes it difficult to provide security to the private network.

3.2.2. Organizational assets

No single site security policy is best for any two organizations. Because different companies have different demands and can take different levels of risk, every security policy is developed for a particular organization. The security policy must be based on carefully conducted security analysis, organizational assets identification, risk analysis, and business risk analysis for that organization.

There are many factors in developing a security policy. Organizations must know what they are trying to protect, what they are protecting it from and what are possible threats against organizational assets. One of the most important decisions in developing a security policy is how much security to put up. This will depend on the importance of data being protected because data of different value for an organization will needn different levels of protection. Also there is a trade off between how much security to put up on one hand and the expense of the security solution on the other.

Every organization needs to perform classification of data. This means it has to define the relative value of various types of data used within the company. This evaluation of information can range from low value for information made available to the public, to

high value such as new research results, investment information and other sensitive information.

There are three characteristics that should be considered when trying to protect important data:

- Secrecy which helps with keeping important data private
- Integrity ensures that only authorized personnel can make changes
- Availability is concerned with providing continual access to some data

Besides data there are other resources of an organization that might also need protection. These resources include company's hardware, software, documentation, etc. Intruders can often use computer time and disk space without making any damage to a company's data and other equipment. But an organization spends money on those resources and it has every right to use it whenever and however it wants. Thus, one of the first steps in developing security policy should be creating a list of all items that need to be protected, and then establishing procedures and rules for accessing resources located on the company's private network.

3.2.3. Development of a security policy

A security policy should be captured in a document that describes the organization's network security needs and concerns. Creation of this document is the first step in building an effective network security system. Policy creation must be a joint effort of many groups. It should be formulated with and have support from top management which will have the power to enforce the policy and technical personnel which will advise on the implementation of the policy .It must be clear that every misunderstanding or conflict between groups that are included in producing the security policy can lead to security problems (so-called security holes).

This effort should end with an issued security policy that covers such things as:

- Network service access - defines services which will be allowed or disallowed from the private network, as well as ways in which these services will be used.
- Physical access - physical security of the place where hardware, software or communication circuits reside must be adequate, and identification of authorized personnel that can enter those otherwise restricted areas.

- Limits of acceptable behavior – effort should be made to inform the users about what is considered proper use of their accounts; this can be done by an educational campaign or by giving the users a policy statement.
- Specific responses to security violations – security policy should establish a number of predefined responses that should be taken in case of violation, to ensure prompt and proper enforcement.
- Reviewing of the policy – the policy should be reviewed on a regular basis; responsibility for maintenance and enforcement of the policy should also be defined; this can be individual or committee responsibility.

Developing a security policy should be only one part of the overall security efforts. Equally important is education of users. The site security policy should include a formalized process, which communicates the security policy to all users. Personnel who are responsible for administering the network should make users advised of how computer and network systems are expected to be used. Users should understand how common security breaches are and how costly these breaches can be.

3.3. Authentication

One of the fundamental issues involved in network security is that access to valuable resources must be restricted to authorized people and processes. Authentication is the process of determining the accuracy of the user's claimed identity. The user authentication system attempts to prevent unauthorized users from gaining access by requiring users to validate their authorization to use the system .

A closely related concept is the authentication of objects such as messages. When the content of a message is important, the receiver may find it necessary to be sure of its source and integrity.Data integrity ensures that data have not been altered or destroyed in an unauthorized manner along the way. Similarly, the sender may desire positive proof of delivery. Digital systems provide these necessary authentication mechanisms.

3.3.1. User identification and authentication

The first step in access control is for the individual to present identification and authentication of that identification. Users begin the authentication process every time they log in by entering their user ID. Once they are logged they have to prove their identity or to

authenticate themselves. Passwords that must be presented to the system are the most common form of authentication.

The authentication information must be validated before the user identification isaccepted. Passwords presented by users are compared with previously stored informationassociated with the user identification; a match results in acceptance of the identification.The stored information is commonly the user's encrypted password. This encryptionprotects the authentication information even if the password is disclosed.

A computer system may employ three different ways to verify a user's identity:

- By something they know. This is the most common method where the system requires the user to provide specific information to access the system.
- By something they have. In this case a system requires that a user possess a physical key to access the system.
- By something they are. The third type of identification is a biometric key, which usesthe fact that no two human beings are the same .

Authentication mechanisms must uniquely and unforgeably identify an individual. Possession of knowledge or a thing means that it could be lost, duplicated, or stolen by someone else. To prevent unauthorized users from gaining access by stealing one of the keys, a computer system can use more then one of these techniques. Of course, as we add more types of verification, certainty of authentication goes up, but so does the cost. In real life, a computer system heavily relies on knowledge and possession keys, while biometric keys are too expensive and hence are used only for extreme security requirements.

3.3.1.1. Informational keys

Informational keys are usually passwords, phrases, personal identification numbers (PIN numbers) that an authorized user knows and can provide to the system when requested. Many systems allow the user to create his own password so that it is more memorable. In general, a user's password should be easy to remember but difficult to guess. Unfortunately, there are a number of ways in which a password can be compromised. For example, someone can see the username and password while the authorized user gains access, users can tell their password to a co-worker, or users can write a password down and leave it out in a public place where it can be easily accessed by casual observers or co-workers. To prevent unauthorized users from accessing a computer account

a one-time password can be used. In this case a list of passwords which will work only one time for a given authorized user is generated. Of course, special care should be taken for protecting the password list from theft or duplication.

3.3.1.2. Physical keys

Physical keys are objects that users must have to gain access to the system. They are widely used because they provide a higher level of security than passwords alone. The commonly used physical keys are magnetic-strip cards, smartcards, and specialized calculators .In order to use magnetic cards, a computer system must have card readers. The process of validation begins when the user enters both a card and Access number and it has four stages: information input, encryption, comparison, and logging. The authentication system then encrypts the access number entered by the user and compares it to the expected value obtained from the system. If these values match, the authentication system grants the user access.

Smartcards also contain information about the identity of the card holder and are used in a similar manner. The difference is that smartcards contain a microprocessor, inputoutput ports, and a few kilobytes of non-volatile memory, instead of magnetic recording material, and can perform computations that may improve the security of the card .A calculator looks very much like a simple calculator with a few additional functions. In addition to possessing a calculator, the user has to remember his user name and personal access number. When the user wants to access the computer system it has to provide his user name. The authentication system returns a challenge value back to the user, which then has to enter that value and his personal access number into his calculator. After performing some mathematical computation, the calculator returns a response value to the user. The user then presents the response value to the system, and if the number presented matches the value expected by the system, access is granted.

3.3.1.3. Biometric keys

Biometric keys provide many advantages over types of keys that were discussed so far. The three primary advantages of biometric keys are they are unique, they are difficult to duplicate or forge and they are always with a user. Biometric approach presents the higher technology solution to access control problems, but requires special hardware that

effectively limits the applicability of biometric techniques. Commonly used biometric keys include voice prints, fingerprints, retinal prints, and hand geometry .

3.3.2. Message authentication

Message authentication is the ability of the receiver to verify that the received message is not altered by some attacker, is not a reply of an earlier message sent from an attacker, or is a message completely made up by an attacker. Verification of the source and original content of a message should be applied always when a new message is received. There are three different methods for message authentication:

- Message encryption, where ciphertext of entire message serves for authentication of message.
- Appending a MAC or cryptographic checksum to the message.
- Hash function that maps a message of any length into a fixed-length hash value, which serves as the authenticator.

3.3.2.1. Message encryption

In conventional encryption or so-called symmetric encryption method, a message transmitted from source A to destination B is encrypted using a secret key K shared by A and B. So, if no other party knows the key, we may say that confidentiality as well as some degree of authentication of the message is provided. Symmetric encryption does not provide a signature so the receiver could forge the message or the sender could deny the message .In this method there is mainly the risk that an outsider will find out the secret key shared by the two communicants A and B. The most common symmetric encryption method is DES algorithm.

In the public-key encryption or so-called asymmetric encryption method, the source A uses the public key KB1 of the destination B to encrypt the message, and because only B has the corresponding private key KB2 only B can decrypt the message. This provides confidentiality but not authentication. To provide authentication, A uses its private key KA2 to encrypt the message, and B uses A's public key KA1 to decrypt the message. Because only A could have constructed the ciphertext, B has the means to prove that he message must have come from A. In effect, A has "signed" the message by using its private key, providing what is known as digital signature. To provide both confidentiality and

authentication, A can encrypt the message first using its private key, which provides the digital signature, and then using B's public key, which provides confidentiality .

The most common method, though not a U.S. government standard, for public key encryption is the RSA (Rivest, Shamir, Adleman) technique. In contrast, in 1994 the federal government approved its own standard developed by NSA called the Digital Signature Standard (DSS). DSS provides authentication and data integrity; it doesn't provide encryption .In methods based on asymmetric encryption there is mainly the risk that an outsider makes the receiver B believe that he value of the public key of sender A is something other than KA1.

3.3.2.2. Cryptographic checksum

A cryptographic checksum, also known as a Message Authentication Code (MAC), involves the use of authentication function and secret key. MACs have been suggested as a means of providing confirmation of the authenticity of a document between two mutually trusting parties. When A wants to send a message to B, A generates the fixed-size block of data, known as a cryptographic checksum or MAC, as a function of the message and the key. The MAC is then appended to the message and transmitted to the intended recipient. The receiver then performs the same calculation on the received message to generate a new cryptographic checksum. If the received checksum matches the calculated checksum, the receiver can be sure that the message has not been altered.

One of the most widely used cryptographic checksums, refereed to as the Data Authentication Algorithm, makes use of traditional cryptographic algorithms such as Data Encryption Standard (DES), and relies on a secret authentication key to ensure that only authorized personnel could generate a message with the appropriate MAC. However, several technical difficulties have been identified with both the standard MAC and DES-based checksum approaches. In particular, it is shown that MAC checksum length is inadequate .

3.3.2.3. Hash function

Hash function is a form of message authentication that provides data integrity but not the authentication of the sender or receiver. Hash function accepts a variable size message as input and produces a fixed-size hash value. The function manipulates ("hashes") all the bits of the message in a carefully defined way and appends the hash value

to the message at the source. The receiver authenticates that message by recomputing the hash value. It compares its own result to a table, and if the results match, the data have not been changed between sender and receiver. Depending what is required, hash code can be used in a variety of ways to provide message authentication and/or confidentiality. Popular hashing algorithms include Kaliski's MD2 algorithm, Rivest's MD5 algorithm, and NIST's Secure Hashing Algorith (SAH). SAH is considered the most secure to date .

3.4. Encryption

Encryption plays an important role in the security of computer networks. It can be used to protect data in transit through the communication network as well as data in storage. Encryption or encipherment can be defined as the process of coding of plaintext through an algorithm or transform table into a form so that others cannot understand it - effectively producing ciphertext or a cipher . In order to read the original data, the receiver must convert it back through the process called decryption. To perform decryption, the receiver must possess the key.

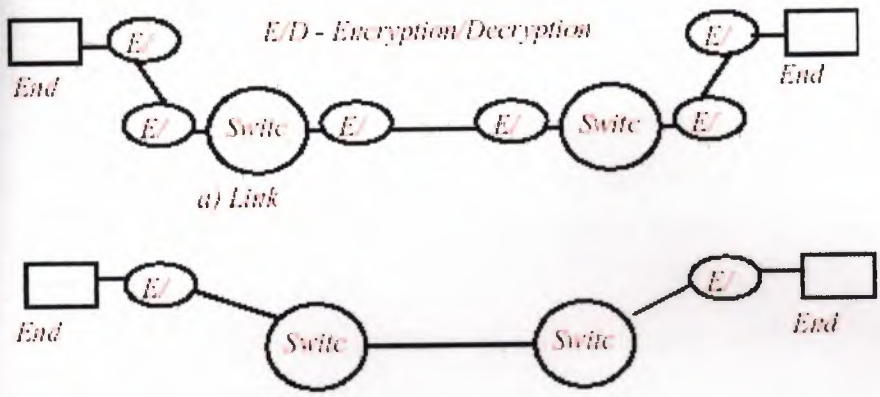
Encryption mechanisms rely on keys or passwords, and the longer the key the more difficult the encrypted data is to break. Also, because each of the encryption mechanisms depends on the security of the keys it uses, management of the keys requires special attention. Key management involves generation, distribution, storage, and regular changing of cryptographic keys.

There are basically two types of encryption methods: symmetric (conventional or onekey) and asymmetric (public or two-key) systems. As we already mentioned the most widely used symmetric method is Data Encryption Standard (DES) which has been adopted as a standard by the U.S. federal government The DES has been implemented in both a software form and hardware form. A public key system differs from symmetric in that it uses different keys for decryption and encryption. RSA encryption technique is the most widely used two-key system, although it is not an U.S. government standard. RSA has proven to be an extremely reliable algorithm used for both public key encryption and digital signatures .

3.4.1. Link encryption

Encryption can be performed link by link or end-to-end. So-called link encryption is described as providing protection for a line with no intermediate nodes. The link encryption

is appropriate for point-to-point circuits. It functions at the physical level where the entire bit stream being transmitted is encrypted . In the case of link encryption, link encryption devices are required between every node (could be a router, bridge, or x.25 switch) and the circuit connected to it (see Figure 3.1.a).



b) End-to-end

Figure 3.1. Internetwork encryption

In the case of a switched network model, the link encryption process may be repeated many times as a series of isolated transmissions as the message transverses a complex network. It is obvious that some of the protocol information, such as addresses or control information in X.25 or TCP/IP networks, must be available to the switch in plaintext in order that it can perform its function. Because information will be in plaintext while in the switch, there are potential security vulnerabilities in the switches such as sourcerouting attacks, RIP-spoofing, and other attacks.

3.4.2. End-to-end encryption

It certainly would be more secure to encrypt at one end, transport all encrypted data transparently to the other end, and then decrypt the information. Expanding encryption into higher protocol layers may be used to secure any conversation, regardless of the number of hops throughout the network. End-to-end encryption is described as encrypting only user data; network data must remain unaltered for intermediate network nodes. In this way, data do not exist in plaintext form at intermediate nodes. The end-toend information is thereby protected, while leaving necessary routing and control information in plaintext. This

approach also saves tremendously on encryption devices and greatly simplifies key management (see Figure 3.1.b).

4. THE FIREWALL CONCEPT

A number of security problems with the Internet mentioned in section 3.1 could be reduced through the use of existing techniques and tools. The most widely known and widely used tool to provide protection against unwanted intruders into corporate networks is the firewall.

A firewall is not simply a set of hardware components such as a router, host computer, or some combination of these that provides security to a network, rather it is an approach to security. It helps implement a larger corporate security policy that defines the services and access to be permitted. Consequently, the various ways of configuring the equipment that compose a firewall system will depend upon a site's particular security policy, budget and overall operation.

There are a number of definitions of a firewall. For example, a firewall can be defined as "a barrier between two networks that is used as a mechanism to protect an internal, often called the trusted network, from an external network, called the untrusted network." A firewall system is usually located at a point at which protected internal network and a public network, such as the Internet, connect (see Figure 4.1.).

The main function of a firewall is to centralize access control at the Internet connection. With this in mind it is clear that a firewall simplifies security management, since network ecurity is consolidated on the firewall system rather than being distributed to every host in the entire private network. It can also be used to completely 'hide' the users on the private network from the external network.

The firewall system is responsible for allowing access for authorized individuals and at the same time for shielding a site from protocols and services that can be abused from hosts outside the private network. Thus rules, specified by the private network administrator, defining authorized traffic should be defined to the firewall and enforced by it. Any traffic not specifically authorized according to these rules must be blocked by the firewall. Of course, for a firewall to be effective, all traffic to and from the Internet must pass through the firewall, where it can be examined. The firewall itself should also be secure and immune to penetration.

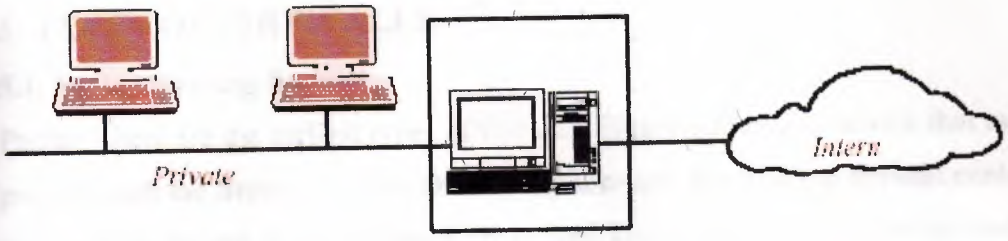


Figure 4.1. Schematic of firewall

Firewall systems can be deployed within private networks as well. In such cases the firewall will protect parts of the internal network from other parts of that same network, rather than from the Internet. This is very useful technology because not everyone in an organization needs access to the same services and data, and some subnets of an organization need a higher level of security. A firewall deployed within a corporate network will prevent unauthorized access to particular subnets, workgroups, or LANs, such as the accounting workgroup, research and development department, etc., from the rest of the network. This is particularly important because many sources claim that 70 percent of all security problems originate from inside an organization.

Today, most firewall systems use one or more of three types of firewall technology: packet filtering routers, proxy systems and stateful inspection. Packet filters are inexpensive, and transparent to the users; proxy systems are more sophisticated and secure, but not transparent to the users; stateful inspection provides full application-level awareness without requiring a separate proxy for every service. We will discuss all of them in more detail in the next chapter.

There are different implementations of firewalls that can be arranged in different ways. Historically there have been two approaches in the firewall security issues. One approach implied that adequate level of security could be achieved using packet filters available in most routers. The other, and more accepted approach in today's world, suggested that packet filtering could be used, but only in conjunction with proxy systems and proper authentication.

5. TYPES OF FIREWALLS

5.1. Packet filtering firewall

Packet filters are the earliest types of firewall. Filtering firewalls require that every packet pass through the firewall device. On the way through, the filtering firewall controls what data can flow to and from a network. It is well known that routers can be used as filtering firewalls.

This is a good way to establish a packet filtering firewall because a router is necessary to establish the local (LAN) to wide area network (WAN) connection, and it already dealing with the routing of packets. A router may be a dedicated piece of hardware that has no other purpose, or it can be a piece of software that runs on general purpose UNIX or PC system. A normal router (router that doesn't act as a packet filter) has to make a routing decision about each packet it receives; it has determine how to forward a packet towards its destination. In addition to this, the packet filtering router also has to make a decision of whether it should forward that packet or not. The packet filtering router is able to make these decisions according to the security policy, which is implemented through packet filtering rules.

5.1.1. How packet filtering works

Packet filtering is done by setting up filtering rules on a router inserted between the local private and external untrusted network. A firewall implementing packet filtering on a router to operate at the network level is sometimes also referred to as a screening router.

A screening router works at the IP layer (which corresponds to the network layer of OSI protocol architecture). Each IP packet contains source and destination IP addresses, as well as TCP or UDP source and destination port numbers. The firewall checks each IP packet against the filter rules as they pass between the router's interfaces, and accordingly allows or blocks certain types of packets. The more attributes the filtering rules can check on, the better. Usually a screening router can filter IP packets based on the following attributes :

- Source IP address
- Destination IP address
- TCP/UDP source port
- TCP/UDP destination port

- TCP flags
- The IP protocol (whether the packet is a TCP, UDP, or ICMP packet)
- ICMP message type

Adding TCP or UDP port filtering to IP address filtering results in a great deal of flexibility because servers for different services usually reside at a specific port [37]. Not all packet filtering routers currently filter the source TCP/UDP port, but more vendors are starting to incorporate this capability. In addition, the router has knowledge of two more things that are not connected with the format of IP packet, but still can be used as an additional filtering criterion:

- The interface where the packet arrived (secure or insecure network interface)
- The interface where the packet will go out Usually, the packet filtering routers allows users to build a table of permit/deny entries where each line in the table contains some or all of above mentioned criteria. In addition each entry contains an indication of whether packets that match the description are to be allowed or dropped.

5.1.2. What services to filter?

There are two general forms of packet filtering: filtering by address and filtering by service. In the first case some sites might want to block connections from certain addresses such as from hosts or sites that it considers being untrustworthy. Packet filtering rules are then based on the source or destination address, and they don't have to consider what services are involved. This type of filtering is not in use as much, and it serves mainly in blocking incoming packets with forged source addresses.

To use the other form i.e. filtering by service, an organization first needs to decide what services it wants to allow or disallow. The decision to filter certain services should already have been defined and driven by the organization's own security policy. There are some services that are inherently vulnerable to abuse and should be blocked at a firewall. For example:

- Trivial File Transfer Protocol (TFTP) usually listens to port 69; TFTP is used for booting diskless workstation; there is no need for booting diskless systems across the Internet and consequently there is no reason to allow TFTP across the firewall.

- The BSD "r" commands such as rlogin – port 513, and rsh – port 514 are used for convenient remote access and if improperly configured can permit unauthorized access to accounts; it is safer to use alternative protocol such as Telnet, FTP, etc.
- Talk is a text-based real-time conferencing system between two people; talk servers use either port 517 or 518; it is not possible to safely filter talk.
- Network File System (NFS) currently uses the port number 2049; NFS server relies on the IP address to check whether or not the client is allowed to access that file system making it vulnerable to address forgery; it is not recommended to allow NFS across the organization's firewall.
- Network Information Service / Yellow Pages (NIS/YP); NIS/YP servers do not use predictable port numbers so it cannot be adequately handled neither with the packet filtering system nor with proxies; it is not recommended to allow NFS across the organization's firewall.

Other services are usually allowed but restricted to only those systems that need them. Blocking all of these services would cripple the access to the Internet and its unlimited resources. These services are:

- SMTP - server listen on port 25 for incoming SMTP connections; packet filtering rules should be used to restrict SMTP connections from external hosts to only bastion host, and from bastion host to a internal mail server, while all internal users should be allowed to send outgoing mail to the bastion host.
- FTP - server uses port 21 for command channel, and port 20 for data channel. There are two modes of FTP connection supported by server and client: normal mode and passive mode. Packet filtering could be used to allow incoming FTP connections to the organization's bastion host. If FTP client supports passive mode then outgoing FTP connections could be allowed via packet filtering as well, but if doesn't then it is better to use an FTP proxy server.
- Telnet - server listens on port 23. It is recommended to permit incoming Telnet session only to specific hosts and all outgoing Telnet traffic, both of which can be allowed via packet filtering.
- NNTP - port 119; packet filtering or proxying should be used to only allow connections from trusted external NNTP server to a local news server.

- DNS - port 53; it has some security problems such as revealing too much information that can be useful to attackers; there are two approaches to set up DNS services: with and without hiding information, depending upon the sensitivity of an organization's data.
- RIP - port 520; it is the oldest routing protocol and it can be spoofed to redirect some packets.
- Gopher and HTTP - ports 70 and 80; HTTP should be restricted to run on dedicated bastion host only .

5.1.3. A few rules for filtering by service

Once the decision about which services an organization wants to allow is made, it is very important how these services would be translated into a particular set of rules for the router, since the router works at the IP level and thus understands and works only with packets. In addition, the selected screening router should allow specification of those rules based on any of previously mentioned attributes. Also it is important that the router applies rules in a predictable order; the simplest order is order specified by the user.

When planning packet filtering rules, it should be kept in mind that protocols are usually bi-directional, which means that one side is sending a request, and the other side is sending a response. So in order to allow some service, the filter on the router should allow packets from that service in both directions. For example, if a user from the internal network wants to retrieve a file with FTP protocol from the external FTP server, it should be allowed to send the request to the external FTP server, but also it has to be allowed to accept a response from that server.

As we said the router works with packets only, so it is important to distinguish between inbound and outbound service, and incoming and outgoing packets . Our FTP example above was an outbound service, but contained both outgoing packet (request to the external server) and incoming packet (respond from the external server).

Also for all services that are based on the TCP protocol, rules for filtering of packets can be based on the TCP flags. As we recall from paragraph 2.4.1, TCP uses a three-way handshake to establish and close connection. The TCP flags are used in such procedure to indicate the type of TCP packet. Although screening routers have capabilities to filter on any of the TCP flag settings, the flags that are most frequently used are the SYN and ACK

flags. The three possible combination of SYN and ACK flag settings for opening of TCP connection are given in the table below:

SYN flag	ACK flag	Meaning
1	0	Open connection
1	1	Acknowledgement of open connection
0	1	Acknowledgement – connection has been established

5.1. Filtering based on the TCP flags

5.1.4. Protocol specific issues for filtering Telnet traffic

If an organization wants to allow outbound Telnet service from its internal network to the external network, packet filtering rules should specify which outgoing and incoming packets are permitted. In the example from Figure 5.2 outgoing packets for specific host with IP address 192.168.2.1 will contain:

- The IP source address (internal for private network) - 192.168.2.1
- The IP destination address (external for private network) - 192.168.1.2
- IP packet type – TCP
- The TCP source port - Telnet client uses random number greater than 1023
- The TCP destination port - Telnet servers use well-known port 23 First outgoing packet which will establish the connection will have SYN bit set, while all others will have ACK bit set.

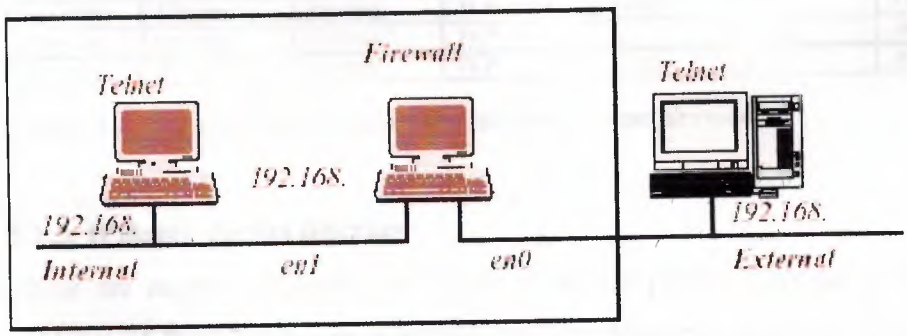


Figure 5.2. Representation of Packet filtering for Telnet traffic

The Telnet server will respond back to the client on a private network and incoming packets will have the following characteristics:

- The IP source address (external for private network) - 192.168.1.2
- The IP destination address (internal for private network) - 192.168.2.1
- IP packet type – TCP
- The TCP source port – port 23
- The TCP destination port - same as source port for the outgoing packets (>1023)
- All incoming packets will have ACK bit set.

Thus, if outbound Telnet service is needed from any host on private network (192.168.2.0) to an external Telnet server, the characteristics can be summarized in the following table:

Packet direction	*Source address*	*Destination address*	*Packet type*	*Source port*	*Destination port*	*Flags*
Outgoing	*192.168.2.0*	*192.168.1.2*	*TCP*	*>1023*	*23*	*SYN*
Incoming	*192.168.1.2*	*192.168.2.0*	*TCP*	*23*	*>1023*	*ACK*
Outgoing	*192.168.2.0*	*192.168.1.2*	*TCP*	*>1023*	*23*	*ACK*

Figure 5.3. Packet characteristics for outbound Telnet service

For the same organization to allow inbound Telnet service, in which users external to the private network communicate with a local Telnet server, packets in general should have the following characteristics:

Packet direction	*Source address*	*Destination address*	*Packet type*	*Source port*	*Destination port*	*Flags*
Incoming	*External*	*Internal*	*TCP*	*>1023*	*23*	*SYN*
Outgoing	*Internal*	*External*	*TCP*	*23*	*>1023*	*ACK*
Incoming	*External*	*Internal*	*TCP*	*>1023*	*23*	*ACK*

Figure 5.4. Packet characteristics for inbound Telnet service

5.1.5. IPRoute packet filtering

There are number of tools that allow us to add packet filtering to PC or UNIX systems. Many of these tools are available for free downloading from the Internet and resources can be found in. The exact mechanism for specifying packet filtering rules varies from product to product. In order to provide a detailed example for setting up filtering rules we chose the IPRoute software package.

IPRoute is a program written by David F. Mischler that runs on a 286 or better CPU. It is intended to be useful for connecting a LAN to an Internet Service Provider, or for routing between LANs [38]. IPRoute has the capability to route IP packets between network interfaces on PC hardware, and besides others it provides IP packet filtering features. A filtered interface has two separate lists of filtering rules: one for incoming packets, and one for outgoing packets. As packets enter or leave the router on a filtered interface they are checked against each filter rule in the order the rules were specified until a match occurs, or the end of the filter list is reached. When a match occurs the action specified in the filter rule is performed on that packet. Packets that do not match any filter rules will be silently dropped; this assumes that a default 'deny' stance from the security policy is in use.

Requirements to build IProute packet filter device are as follows:

- 286 or better PC computer
- two or more ethernet cards
- IPRoute software

IPRoute's commands for setting up packet filtering rules require specification of the interface on which the rule is to be applied, and whether the rule applies to incoming ('in' refers to packet entering the router from outside) or outgoing packets ('out' refers to packet leaving the router) on that interface. IPRoute also has the capability to drop the matching packet and to send an ICMP destination unreachable message back to the packet's originator if action 'deny' is specified, or to silently drop the packet if 'drop' action is specified. Besides the possibility of filtering packets based on the source and destination IP addresses, subnetmask, and TCP/UDP port numbers, IPRoute allows us to filter on the protocol type and flags (SYN or ACK). For example, to allow outbound Telnet service from the above example with IPRoute we could specify:

filter en0 permit out tcp-syn 192.168.2.0/24 192.168.1.2/24:23

filter en0 permit in tcp-xsyn 192.168.1.2/24:23 192.168.2.0/24

filter en0 permit out tcp-xsyn 192.168.2.0/24 192.168.1.2/24:23

where en0 is an external interface on which the rule is to be aplied, tcp-syn indicates an attempt to open a new connection (SYN bit is set), and tcp-xsyn indicates an existing connection (both SYN and ACK bits set or only ACK bit set); /24 indicates the width of the

network mask in bits and :23 specifies port number which is separated from the address part by a colon. When we want a rule to match all addresses on a given network '*' character can be used. Our filtering rules will then be:

filter en0 permit out tcp-syn * 192.168.1.2/24:23

filter en0 permit in tcp-xsyn 192.168.1.2/24:23 *

filter en0 permit out tcp-xsyn * 192.168.1.2/24:23

Appendix A contains example IPRoute configuration with a detailed description of how packet filtering rules should be specified using IPRoute software for two different policies:

- All inbound traffic is forbidden
- FTP, Telnet and Daytime services are, with some restriction, allowed

Contents of the script files, log files, and examples of FTP session for these two cases are also given in appendix A.

5.2 Proxy systems

A more secure and sophisticated type of firewall technology is the proxy system. The Proxy system is usually used in order to impose more control on what is happening at the application layer and because of this proxies are sometimes referred to as applicationlevel gateways . The proxy system is a host running special purpose written code for specific applications. Sets of code are called proxy services and exist both as clients and servers within the physical gateway. The proxy system acts as a server to receive the incoming request and as a client when forwarding the request (see Figure 5.5). If a request is approved and the session is established, the proxy system contacts the real server on behalf of the client (thus the term "proxy") and provides relay of connection between the client and the real server.

Because a proxy firewall doesn't permit any IP packets from the Internet to show up directly on the internal network, a proxy system allows an organization to implement a much stricter security policy than with a packet filtering router. The proxies can also support and perform some additional functions such as user authentication, extra verification, and logging before carrying out the user's intended connection to the application beyond the firewall. Because of these characteristics, the proxy system is considered as one of the most secure types of firewall.

The proxy firewall usually sits between the Internet and a private network. As in packet filtering firewalls there shouldn't be any other connections between a local network and the Internet except for the proxy server that runs user desired applications. Because the proxy server is a main point of contact for users on local networks, and is directly exposed to the possible attacks form the Internet, it has to be specifically secured against those attacks. Consequently, a proxy system is often referred to as a "bastion host". When configured with two network interface cards, one for each required network connection, the bastion host is also called a dual-homed host.

5.2.1. Bastion host features

Because a bastion host is especially vulnerable to attacks from the Internet it has to be specifically protected. There are a few features that can help in providing security for a bastion host:

- First of all it should be clear that users are allowed to access proxy services, but they should never be permitted to log in to the bastion host. If users are allowed to log in to the bastion host, security of the firewall could be threatened.
- As we mentioned above only services that are needed for local users are installed on the bastion host. If some service is not installed then it simply is not available and cannot be attacked.
- It is also necessary to have only a few services ported at the bastion host because complexity builds quickly as applications are added. Bastion hosts that have smaller number of services are simpler and hence more secure.
- Usually authentication is performed on the bastion host before the user is allowed access to the proxy service.
- Bastion hosts supports logging which is one useful tool for discovering attacks. Usually all information about connections are maintained by logging all traffic.
- It is also important to note that each proxy code is independent of all proxies on the bastion host, so that if one has some security related problems it can be removed without consequences to the other application.

5.2.2. How a proxy system works

Unlike packet filtering routers, where direct echange of packets between the internal and external network is allowed, the proxy server does not allow direct flow of IP packets

between the two interfaces through the kernel. When users wants to access some service on the Internet they have to communicate with the proxy system rather than with the real server that offers the desired service. The proxy then acts as both server and client; when receiving an incoming request, it acts as a server, and when forwarding the request, acts as a client (see Figure 5.5). It is the same for the external users who want to connect to the internal servers; they would have to connect first to the proxy system, and then to the destination host.

For example, a user on the local network who wants to use Telnet to connect to a Telnet server on an external network would have to:

- First connect to the proxy, instead of connecting to the final destination computer, and to enter desired external server.
- The proxy will then check if the user is allowed access to the Internet or not based on the variety of criteria specified for that proxy, and accordingly, will make a decision to accept or reject the user connection.
- If the requirements are successfully met, the proxy makes a Telnet connection from the bastion host to the external host.
- Finally, the proxy passes the packets through the other network interface on the bastion host and on to the Internet.

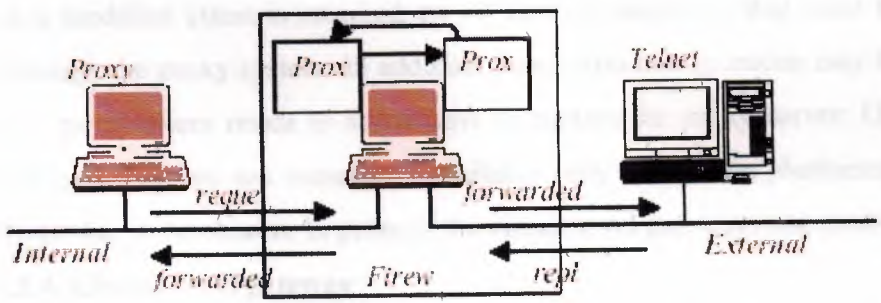


Figure 5.5. Proxy system

It is obvious that the proxy server allows through only those services for which there is a proxy code installed. If the proxy code for a particular application is not installed, the service is not supported by the proxy server and cannot be forwarded across the firewall. For the example above, if the proxy server didn't contain a proxy code for Telnet,

that service would be completely blocked and users wouldn't be able to make connection to the Internet host.

5.2.3. Custom user procedures vs. custom client

As Figure 5.5 shows proxy service is composed of two components: a proxy server and a proxy client. Also, from the previous paragraph (5.2.2.) we can see that a proxy system requires a modified user behavior. The user has to connect to the proxy server instead of connecting to the real host on the Internet. With this approach, users would still be able to use standard client software. This approach has a major drawback in that the users have to learn a custom procedure to follow. Moreover, when a user connects to the Proxy server, it has to specify not only its user name, but also the name of the real server he wants to connect to. This could be a problem because not all clients allow the users to type both user name and host name. Obviously a custom client procedure places some limitation on which clients can be used. Usually, application level gateways use modified procedures.

Another approach is a modified proxy client. This proxy client is a special version of a normal client that is capable of talking to the proxy server. The custom client should be capable of specifying to the proxy server which real server it has to connect to. A modified proxy client can make the firewall transparent to the users by permitting them to specify the real server to which they want to connect. In this case the user will have the illusion that they are connected directly to the real server on the Internet. User behavior would stay the same, but a modified client is required on all internal machines that want to access the Internet through the proxy system. In addition some extra configuration may be necessary, because the proxy client needs to know how to contact the proxy server. Unfortunately, appropriate proxy clients are sometimes available only for certain platforms, so that the right software has to be chosen. In general, the circuit level gateways use modified clients.

5.2.4. Circuit-level gateway

Proxy systems generally fall into two types: application-level gateways and circuit-level gateways. So far we have been mostly concerned with the application-level gateway type of firewall which is a collection of application proxies for each of the separate services used. An application level gateway understands and interprets the commands in the application protocol.

The circuit-level gateway provides relay capabilities in a generalized form that is not limited to a specific application. A circuit gateway simply relays TCP connections (effectively creating a circuit) between the client on the local network and the server on the external network, without interpreting the application protocol, or performing any additional packet examination or filtering. Although the gateway will not typically examine the data, it can keep a log of the amount of data relayed and its intended destination.

The user requesting the service connects to a TCP port on the gateway. The gateway then connects to the destination on the other side of the gateway. After the session is established, the gateway's relay program copy the bytes back and forth: the gateway acts as a wire. For example, when a workstation on the internal network connects to the SMTP port on the gateway, the gateway opens a matching socket on the connection to the Internet and then just transports data between the two connections.

When the connection request is made, the gateway either makes the connection if this is an allowable transaction, or if it is not the connection is not made and an error message can be returned. Sometimes, if used often, a circuit connection can be made automatic for specific network functions. At other times the gateway will need to be told the desired destination and service. Although the circuit-level gateway is usually thought of as a relay for TCP traffic, it can also be used for some UDP applications where a virtual circuit is assumed. In general, circuit-level proxies are often used for outbound traffic in the systems where the internal users are trusted.

A circuit level proxy is a more flexible and general approach to building a proxy server. Because a circuit level proxy can be adapted to serve multiple protocols, it is also called a generic proxy server. One of the disadvantages of the circuit level proxy servers is that it controls connections on the basis of their source and destination and cannot easily determine whether the commands going through it are safe [20]. The other big problem with circuit level gateways is the need to provide new client programs, which can be a difficult task because appropriate client programs are often available only for certain platforms.

5.3. Socks

One of the ways to do proxying is using the SOCKS protocol. SOCKS is an open, industry-standard protocol advanced by the Authenticated Firewall Traversal working

group of the Internet Engineering Task Force (IETF). SOCKS is a very robust circuit level gateway firewall. It was designed to allow TCP-based applications to traverse firewalls in a secure and controlled manner. SOCKS enables easy conversion of existing client/server applications into proxy versions of those same applications.

SOCKS establishes a secure proxy data channel between two computers in a client/server environment. The application client makes a request to SOCKS to communicate with the application server. SOCKS then establishes a proxy circuit to the application server and relays the application data between the client and the server. From the client's perspective SOCKS is transparent, while from the server's perspective SOCKS is a client.

With SOCKS there is no need for a special application server on the firewall, nor do the users need to perform double connections. However the user does have to use a specified version of the application client that is SOCKS aware, and there should be a generic SOCKS server to allow the user's intended access. SOCKS is an example of the Proxy system that requires a custom client, because it requires a change to all existing clientbased software to use the SOCKS libraries, a process known as "socksifying".

- The SOCKS package includes the following components:
- The SOCKS server, which runs on UNIX system
- The SOCKS client library for UNIX system
- SOCKSified version of several standard UNIX client programs

The current SOCKS specification is version 5, which is a backward compatible with previous versions and has multiple enhancements. SOCKS 5 adds key features such as authentication and authentication method negotiation, message integrity and privacy, and UDP proxy to the old SOCKS functionality.

There are other excellent software packages publicly available for proxying. For example the Trusted Information Systems has an Internet Firewall Toolkit (TIS FWTK) that includes a set of individual proxies for the most common Internet services, such as FTP, Telnet, rlogin, HTTP, and others. SOCKS and TISFWTK run on UNIX system, but there are proxy server tools for Windows 95/98/NT available as well, such as WinGate, Spaghetti Proxy Server, Internet Gate, NetProxy, SyGate, etc. Currently all of these packages are freely available on the Internet.

5.3. Stateful multi-layer inspection

Stateful multi-layer inspection (SMLI) is the "third generation" of firewall technology. It was invented and patented by Check Point Software Technologies. Stateful inspection architecture is unique in that it understands the state of any communication through the firewall machine, including packet, connection and application information. Packet filters do not track application or connection state, while application proxies track only application state, not packet or connection state. SMLI examines each packet and extracts relevant packet, communication, and application state information. Extracted state information is then compared against known states (i.e. bit patterns) of "friendly" packets.

The inspection module resides in the operating system kernel, below the network layer, at the lowest software level. By intercepting and inspecting communications at this level, the module can analyze all inbound and outbound packets before they enter the operating system of the gateway machine, ensuring that the operating system is protected from untrusted communication. Only packets that the inspection module verifies to comply with the organization's security policy are processed by the higher protocol layers. The inspection module understands any protocol and application.

In order to determine whether packets comply with the enterprise security policy, the inspection module examines IP addresses, port numbers, and other information. In addition, it analyzes state information from previous communication and other applications, and then stores these state and context information in dynamic state tables. State tables are kept in the operating system kernel memory and cannot become corrupted like disk files. This way firewalls that use SMLI are capable of remembering the state of each ongoing conversation across it. Tables are continually updated, providing cumulative data against which the inspection module checks subsequent communications. If the system fails due to hardware or software error, new tables are allocated and no old or corrupted data is valid anymore.

The inspection module maintains complete security even for connectionless protocols such as UDP. For this protocol the inspection module extracts data from a packet's application content and stores it in the state connection tables, providing context in cases where the application does not provide it [14]. Stateful inspection provides full application-layer awareness without requiring a separate proxy for every service to be

This results in complete transparency to the users, scalability and the ability to

6. BENEFITS AND LIMITATIONS OF FIREWALLS

6.1. Benefits of firewalls

One of the main advantages of the Internet firewall is that it allows the organization to define a centralized "choke point", which means that it forces attackers to use only one access point to the protected network (of course the firewall has to be the only connection between the protected site and the Internet). Because all traffic has to go through the firewall, intrusions from the Internet have to come through the firewall as well, which should be specifically protected against such attacks. Without a firewall, each host on the private network would be exposed to attacks from the Internet.

A firewall simplifies security management because it offers a convenient point where the traffic from and to a protected network can be monitored, and if an attack occurs, an alarm can be generated. A firewall is the best place to audit or log Internet usage. Using this log, a system administrator can track down attempts to bypass security. Because the Internet doesn't have enough registered IP addresses to offer to users anymore, some organizations have to deploy Network Address Translator (NAT) that can help to overcome this problem. Many firewall products have the NAT feature already incorporated into them.

6.1.1. Benefits of packet filtering routers

The primary advantages of packet filtering are fast performance, flexibility, and transparency. The packet filtering router does not require specialized user training or cooperation. The end users are unaware of the presence of the firewall and they can use their standard client programs. Packet filtering routers offer minimum security but at very low cost. Low cost comes from the fact that packet filtering capabilities are available in many hardware and software routing products, both available commercially and freely over the Internet. They can be an appropriate choice for a low risk environment.

6.1.2. Benefits of proxy systems

There are many benefits to the deployment of proxy systems as well. The system administrator has complete control over which services are allowed, since the absence of the proxy for a service means that the service is completely blocked. The firewall can be configured to hide host names and IP addresses behind the firewall, so that all hosts outside the local network see only the gateway. Proxy systems can be used to enforce authentication that will reside only on the gateway, lowering the importance of the internal

host security. Proxies provide superior logging capability at the application level. Finally, the filtering rules are much simpler for a proxy system than for a packet filtering router.

6.2. Limitations of firewalls

The fact that all the proposed security of the system is based on the security of the firewall is also its weakness. Because of that it is important to have the firewall correctly administrated. One open breach and an intruder can attack whatever system he wants.

Another limitation of the firewalls is that they cannot protect against attacks that do not pass the firewall. A centralized choke point that an organization had in mind to establish with the installation of the firewall is useless if there is an effective way for an attacker to go around it. For example, there can be dozens of unsecured dial-up lines from a protected network that can be attacked easily. These types of connections should be forbidden by the organization's security policy, and users should know that they are not allowed to get their own connection to the external world.

Firewall systems cannot protect an organization from traitors and inside spies that have their own passwords and access to private network resources, nor from outsiders who stole passwords from legitimate users. They can easily copy sensitive information onto floppy or zip disks and take them out from an organization.

6.2.1. Limitations of packet filtering routers

In addition to previously mentioned common limitations to all firewalls, packet filtering routers have a disadvantage that packet filtering rules become long and complex quickly, making it difficult to manage and thus reducing overall security. Also packet filtering rules are very difficult to get right, because people do not usually think in terms of packets, IP addresses or port numbers. Not only are packet filters difficult to configure correctly, but also they are easy to get wrong allowing unintentional access to the private network. Once configured it is hard to test rule implementations. Another limitation is that some protocols are difficult or impossible to allow safely with packet filtering only. Packet filtering routers provide little or no useful logging, and strong user authentication is not supported with some packet filtering routers.

6.2.2. Limitations of proxy systems

Probably the greatest limitation of the proxy systems is that they either require users to use modified clients (for each of the services that users need separate software should be installed), or may force users to change their normal work pattern by adding steps when making the connection. Another difficulty is when a new service of interest for an organization is not supported by a proxy. In such cases an organization has to deny the service until the firewall vendor develops a secure proxy for a particular service. Clearly, new services may not be introduced to an organization's users on a timely basis. Also the proxy systems are more expensive than packet filtering routers.

7. FIREWALL ARCHITECTURE

7.1. Introduction

Having introduced the principles underlying the packet filter and proxy systems, we can now observe how these components can be configured to build an effective Internet firewall system. Those components can be used either alone or together, and there is a lot of flexibility in how they can be combined. It is important, though, that potential benefits and drawbacks of possible architectures are explored before they are implemented. There is no single correct answer for the design and deployment of Internet firewalls for every organization. Only after making decisions about the security policy, the technical background of their staff, budget issues, and possible threat of attacks, can the organization make a decision about specific components of its firewall systems.

Although there is a lot of variation in architectures, the most common are:

- Dual-homed hosts
- Screened hosts, and
- Screened subnets

7.2. Dual-homed hosts

Dual-homed host is a TCP/IP term that refers to a host with two network interface cards (NICs), one for each required interface. Each NIC is connected to a network and has its own IP address, as shown in Figure 7.1. The dual-homed host could act as a router between the networks these interfaces are attached to. However, to implement a dualhomed host type of firewall architecture, the host's IP routing capability should be disabled. If the IP forwarding capability is disabled, the host can provide network traffic isolation between these two networks it connects to. Systems on both side of the firewall can communicate with the dual-homed host, but there is no exchange of network traffic between these two systems. Because dual-homed hosts allow absolutely no access to internal networks, they provide a very high level of control.

A dual-homed host can provide access to network services only by proxying them or by having users log into the dual-homed host. Dual-homed hosts that do not use Proxy services require users to have accounts on the gateway for access to the Internet. This is not

recommended and can present security problems by itself, as having multiple user accounts on a firewall can lead to users' mistakes and consequently to intruders' attacks.

Allowing access to the Internet services on the dual-homed host is less problematic and safer with setting up proxies. This type of firewall implements the following security stance 'all services that are not specifically permitted are prohibited', since no services can pass the dual-homed host except those for which proxies are established. This approach has the same disadvantages as proxy systems, i.e. proxies may not be available for all services an organization might be interested in.

To increase protection of the private network two-stage security can be established. In addition to an application-level gateway, a packet filtering router can be placed between the Internet and the private network. The network between the packet filtering router and the gateway is called a screened subnet. On the screened subnet are usually placed information servers such as e-mail, Gopher, or WWW machines that are open to outside users (Figure 7.1.). This ability of the screened subnet to isolate traffic concerned with an information server from the other traffic of the site, adds to security because the dualhomed host would prevent intruders from further attacking site systems, although they could possibly break into the information server.

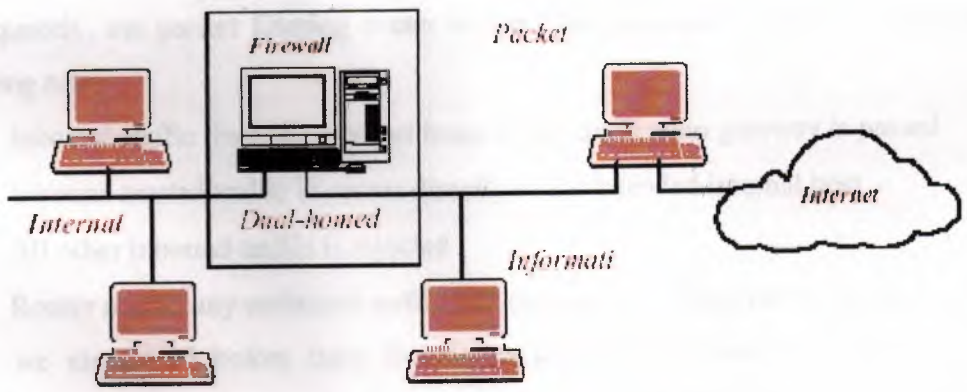


Figure 7.1. Dual-homed host firewall with router.

7.3. Screened hosts

The screened host firewall combines a packet filtering router and an application gateway which has only one network interface. The packet filtering router is placed between the internal and external network as a first line of defense. The application

gateway is configured with only one network interface card that is connected to the internal network (Figure 7.2.). The packet filtering router is configured in such a way that it sends all received traffic from the external network to the application gateway first. Only traffic that passes filtering rules imposed by the screening router would be delivered to the application gateway.

However, the screened host firewall can be made more flexible by permitting the packet filtering router to pass certain trusted services directly to the internal network. Configured this way, the screened host firewall is more flexible than the dual-homed host firewall although at some expense to security. The applications that may be considered trusted might be those for which proxy service does not exist or those for which the risk of using such services has been evaluated and found acceptable. For example services such as Network time Protocol, which is considered low-risk could be allowed. It is also fairly common to allow Domain Name Service so that hosts on the inside of the packet fitler can access Internet services.

It is possible to combine these two approaches for different services. Some trusted services may be allowed directly via packet filtering as mentioned above, while others may be permitted only indirectly – they have to pass through the application gateway first. Implementation of a particular service depends on the organization's security policy. Consequently, the packet filtering router has to filter application traffic according to the following rules:

- Inbound traffic from the Internet hosts to the application gateway is passed
- Inbound trusted traffic is passes directly to the intended internal host
- All other inbound traffic is rejected
- Router rejects any outbound traffic that did not come from the application gateway

As we mentioned before rules for the packet filtering router can be complex and difficult to get right. However, in the case of the screened subnet architecture, the router only needs to limit traffic to the application gateway. Because of this, rules for the packet filtering router don't have to be as complex as if the packet filter were used alone.

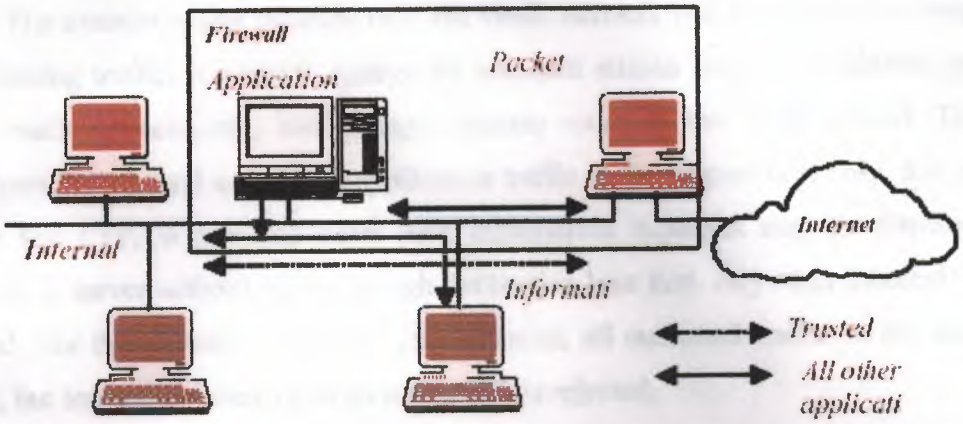


Figure 7.2. A screened host architecture

7.4. Sreened subnet

The screened subnet architecture employs two packet filtering routers and a bastion host. This firewall system creates the most secure firewall system architecture by adding an exterior router to the screened host architecture that further isolates the internal network from the Internet. To break into the internal network with this type of architecture, an attacker would have to get past both routers, meaning that even if the bastion host is breached, the intruder would have to break into the interior router (see Figure 7.3).

In figure 7.3 two routers are used to create an inner, screened subnet. The screened subnet functions as a small, isolated network positioned between the Internet and the private network. Although both the untrusted external network and the internal network can access the screened subnet, no network traffic can flow between them through the screened subnet. This subnet is sometimes referred to as the 'demilitarized zone' (DMZ) network. This DMZ network houses the bastion host, information servers, modem pools, and other public servers.

The external router could be set up to advertise only the DMZ network to the Internet, i.e. the bastion host, information and other public servers would be the only systems known from the Internet. This ensures that the private network is 'invisible' and that it cannot be known to the Internet via routing table and DNS information exchange. Inside routers, on the other hand, advertise the DMZ network only to the private network. Because the systems on the private network do not have direct routes to the Internet, they can Access the Internet only via the proxy services residing on the bastion host.

The exterior router protects both the DMZ network and the internal network from the incoming traffic. It protects against the standard attacks such as IP address spoofing, source routing attacks, etc., and manages Internet access to the DMZ network. The outer router permits inbound e-mail and application traffic to the bastion host only. It is possible though that FTP, WWW and other such information inqueries may go directly to the information server without going through the bastion host first. Any other inbound traffic is rejected. For the outbound traffic is just opposite; all outbound traffic to the Internet is routed, but any traffic intended to an inside host is rejected.

The inside router provides a second line of defense, managing DMZ access to the private network. It permits inbound traffic that originates from the bastion host only. All other traffic, such as FTP and WWW, is directed by the external router to the bastion host or to the appropriate information server. Accordingly, all such traffic on the inside router will be rejected. The outbound traffic is directed only to the bastion host, or possibly to the information server.

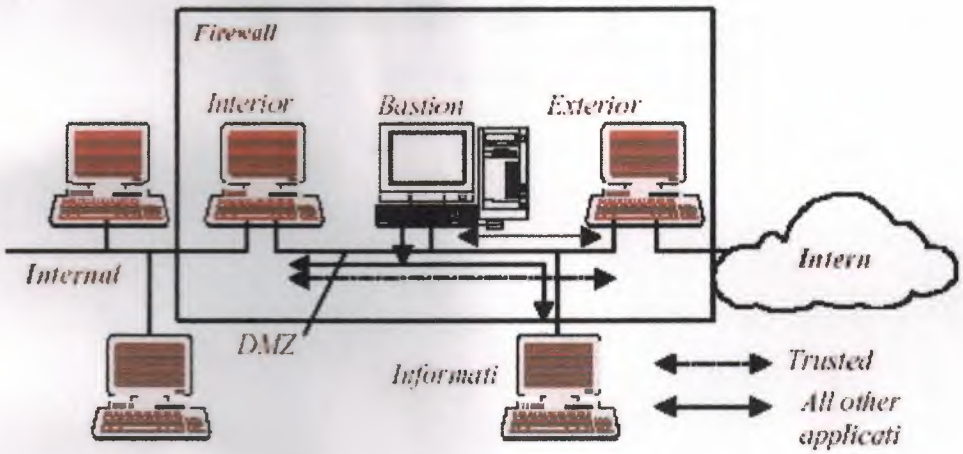


Figure 7.3. Screened subnet architecture

Screened subnets are more secure than screened hosts because of the additional DMZ network. However, screened subnets can be made to allow the same 'trusted' application to bypass the bastion host, thereby subverting the policy. Another disadvantage of screened subnets is their dependence on routers for a large portion of the security

provided. As noted earlier, packet filtering routers are sometimes complex to configure and potential mistakes can open security holes.

Conclusion

Many private networks feel the need to connect to the Internet, so that they can use services and resources of the Internet. There are millions of people who are using the Internet for different purposes and some of then can attempt to break into private computer networks and access remote services that they are not authorized to use. Since the private networks can contain important and confidential data, network security is very important for any organization.

Firewalls are the best way to keep sits secure although one has to include other types of security in the site's overall security. For this reason, the major firewall vendors have incorporated additional security technologies into their firewall products and gone into a partnership with other security vendors to offer complete Internet security solution.

A good security solution should be powerful enough to support the features that the administrator needs, including the capability to inform the administrator of potential security back doors, automatic incident reporting to inform the administrator when a security breach has occurred, and secure management of the firewall itself so hackers cannot reconfigure the firewall and create security problems. Such security technology should also be inexpensive, easy to implement and transparent to end users.

REFERENCES

1. White, G. B., Fisch, E. A. and Pooch, U. W. Computer System and Network Security.

Boca Raton, Florida: CRC Press, Inc., 1996.

2. Abrams, M. A. and Podell, H. J., ed. "Access Control and Authentication." Tutorial:

Computer and Network Security. pp. 349-354, Washington, D. C.: IEEE Computer Society Press, 1987.

3. Simonds, F. Network Security: Data and Voice Communications. New Yourk: McGraw-Hill, 1996.

4. Stallings, W. Network and Internetwork Security. Englewood Cliffs, NJ: Prentice-Hall, 1995.

5. How to Develop a Network Security Policy.

Russell, D. and Gangemi Sr., G. T. Computer Security Basics. Sebastopol, CA: O'Reilly & Associates, Inc., 1991.

6. Hyun-Jung, K. "Biometrics, Is it a Viable Proposition for Identity Authentication and Access Control?" Computers and Security, Vol.14 No. 3, (1992): 205-214.

7. Muftic, S. Security Mechanisms for Computer Networks. Chichester, England: Ellis Horwood Limited, 1989.

8. Bellovin, S. M. "Security Problems in the TCP/IP Protocol Suite." Computer Communication Review, Vol. 19, No. 2, (1989): 32-47.

9. Wack, J. P. and Carnahan, L. J. Keeping Your Site Comfortably Secure: An Introduction to Internet Firewalls. NIST Special Publication 800-10, U. S.

Comer, D. E. Computer Networks and Internets. Upper Saddle River, NJ: Prentice Hall, 1997.

10. Check Point FireWall-1 White Paper. Check Point Software Technologies Ltd.

11. Hendry, M. Practical Computer Network Security. Norwood, MA: Artech House, Inc., 1995.

12. Siyan, K. and Hare, C. Internet Firewalls and Network Security. Indianapolis, IN: New Riders Publishing, 1995.

13. Chapman, D. B. and Zwicky, E. D. Building Internet Firewalls. Sebastopol, CA:

O'Reilly & Associates, Inc., 1995.

14. Feit, S. TCP/IP Architecture, Protocols, and Implementation with IPv6 and IP security. New York: McGraw-Hill, 2997.

15. FitzGerald, J. and Dennis, A. Business Data Communications and Networking. New York: John Wiley & Sons, Inc., 1996.