Network Firewall Technologies

David W Chadwick
IS Institute, University of Salford, Salford, M5 4WT, England

Abstract. This paper provides an overview of the topic of network firewalls and the authentication methods that they support. The reasons why a firewall is needed are given, plus the advantages and disadvantages of using a firewall. The components that comprise a firewall are introduced, along with the authentication methods that can be used by firewalls. Finally, typical firewall configurations are described, along with the advantages and disadvantages of each configuration.

1. Security Threats from connecting to the Internet

Most organisations today have an internal network that interconnects their computer systems. There is usually a high degree of trust between the computer systems in the network, particularly if the network is private. However, many organizations now see the benefits of connecting to the Internet. But, the Internet is inherently an insecure network. Some of the threats inherent in the Internet include:

Weak or No Authentication required. Several services e.g. rlogin, require no password to be given when a user logs in. Other services provide information with no or little authentication e.g. anonymous FTP, and WWW. Other services trust the caller at the other end to provide correct identification information e.g. TCP and UDP trust the IP address of the remote station; whilst other services grant access at too large a granularity e.g. NFS grants access to anyone from a particular remote host. Finally many services require passwords to be transmitted in the clear across the network, which make them vulnerable to capture and replay.

Insecure software. Internet software, particularly shareware, free or low cost packages, often have bugs or design flaws in them usually as a result of poor design or insufficient testing of the software. But due to their ready availability and low cost, many people still take the packages. Examples include: the UNIX sendmail program which has had numerous vulnerabilities reported in it, and a freeware FTP product which contained a Trojan Horse that allowed privilege access to the server. Unscrupulous people are always ready to exploit these weaknesses.

Sniffer programs. In 1994 the CERT reported that thousands of systems on the Internet had been compromised by hackers, and sniffer programs installed on them. Sniffer programs monitor network traffic for usernames and passwords, subsequently making these available to the hacker.

Cracker programs. These programs, widely available on the Internet, run in background mode on a machine, encrypting thousands of different words and comparing these to the encrypted passwords stored on the machine. These so called *dictionary* attacks (because the words are held in a dictionary) are often very successful, providing the hacker with up to a third of the passwords on a machine.

Port Scanners. These programs, again available freely from the Internet, will send messages to all the TCP and UDP ports on a remote computer to see if any of them are open and waiting to receive a call. Once an open port has been located, the hacker will then try to get in to the computer through it.

Ease of Masquerade (Spoofing). The above make it relatively easy for the hacker to exploit the trust inherent in the Internet, or to capture passwords and replay them. Other security weaknesses include: the SMTP protocol uses ASCII messages to transfer messages, so a hacker can TELNET into an SMTP port and simply type in a bogus Email message; a feature called IP source routing allows a caller to falsify its IP address, and to provide the recipient with a return path directly back to itself.

So how can an organization securely connect to the Internet? One solution is to use one or more network firewalls.

2. What is a Firewall ?

A firewall is a secure Internet gateway that is used to interconnect a private network to the Internet (see Figure 1). There are a number of components that make up a firewall:

i) the Internet access security policy of the organisation. This states, at a high level, what degree of security the organisation expects when connecting to the Internet. The security policy is independent of technology and techniques, and should have a lifetime independent of the equipment used. An example of

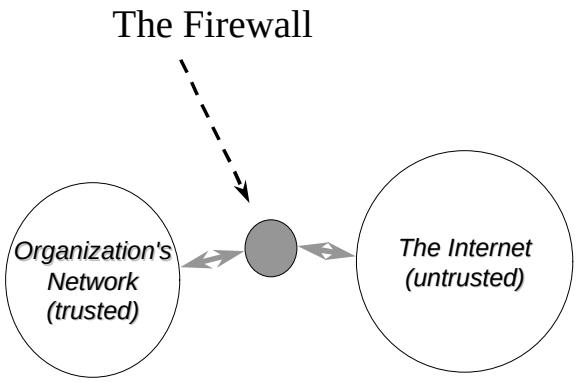

Figure 1

statements from such a security policy might be: external users will not be allowed to access the corporate network without a strong level of authentication; any corporate information not in the public domain must be transferred across the Internet in a confidential manner, and corporate users will only be allowed to send electronic mail to the Internet - all other services will be banned.

ii) the mapping of the security policy onto technical designs and procedures that are to be followed when connecting to the Internet. This information will be updated as new technology is announced, and as system configurations change etc. For example, regarding authentication, the technical design might specify the use of one-time passwords. Technical designs are usually based on one of two security policies, either:

permit any service unless it is expressly denied, or
deny any service unless it is expressly permitted.

The latter is clearly the more secure of the two.

iii) the firewall system, which is the hardware and software which implements the firewall. Typical firewall systems comprise a IP packet filtering router, and a host computer (sometimes called a bastion host or application gateway) running application filtering and authentication software.

Each of these firewall components are essential. A firewall system without an Internet access security policy cannot be correctly configured. A policy without enforced procedures is worthless as it is ignored.

3. Advantages of Firewalls

Firewalls have a number of advantages.

They can stop incoming requests to inherently insecure services, e.g. you can disallow rlogin, or RPC services such as NFS.

They can control access to other services e.g.

bar callers from certain IP addresses,
filter the service operations (both incoming and outgoing), e.g. stop FTP writes
hide information e.g. by only allowing access to certain directories or systems

They are more cost effective than securing each host on the corporate network since there is often only one or a few firewall systems to concentrate on.

They are more secure than securing each host due to:

the complexity of the software on the host - this makes it easier for security loopholes to appear. In contrast, firewalls usually have simplified operating systems and don’t run complex application software,

the number of hosts that need to be secured (the security of the whole is only as strong as the weakest link).

4. Disadvantages of Firewalls

Firewalls are not the be all and end all of network security. They do have some disadvantages, such as:

They are a central point for attack, and if an intruder breaks through the firewall they may have unlimited access to the corporate network.

They may restrict legitimate users from accessing valuable services, for example, corporate users may not be let out onto the Web, or when working away from home a corporate user may not have full access to the organization's network.

They do not protect against back door attacks, and may encourage users to enter and leave via the backdoor, particularly if the service restrictions are severe enough. Examples of backdoor entrance points to the corporate network are:

modems, and importing/exporting floppy discs. The security policy needs to cover these aspects as well.

They can be a bottleneck to throughput, since all connections must go via the firewall system.

Firewall systems on their own cannot protect the network against smuggling i.e. the importation or exportation of banned material through the firewall e.g. games programs as attachments to Email messages. Smuggling could still be a significant source of virus infection if users download software from external bulletin boards etc. The recent Melissa and Love Bug viruses were smuggled inside Email messages unbeknown to the recipients. This is an area that the security policy needs to address. There are software packages that can help in this e.g. Mimesweeper runs in the firewall and will check Email attachments before letting them pass. It will remove potentially dangerous attachments or stop the Email altogether.

The biggest disadvantage of a firewall is that it gives no protection against the inside attacker. Since most corporate computer crime is perpetrated by internal users, a firewall offers little protection against this threat. E.g. an employee may not be able to Email sensitive data from the site, but they may be able to copy it onto a floppy disc and post it.

Consequently organizations need to balance the amount of time and money they spend on firewalls with that spent on other aspects of information security.

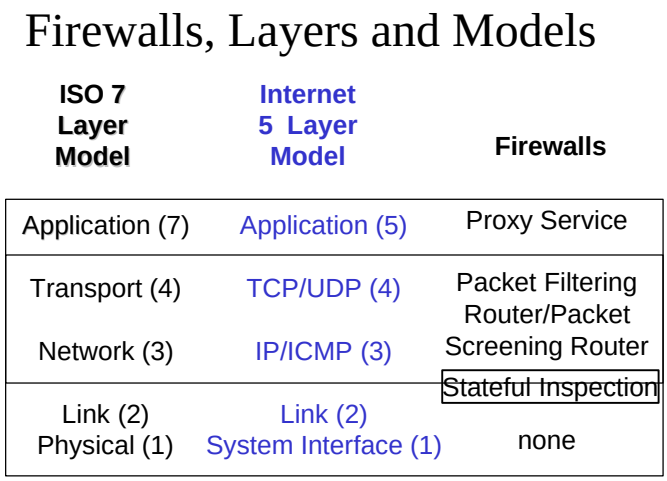

Figure 2

5. Models, Layers and Firewalls

ISO uses a 7 layer model for Open Systems Interconnection, whereas the Internet can be regarded as having a 5 layer model. Whereabouts in these models are firewall systems placed?

Firewall systems are usually placed at layers 3, 4 and 5 of the Internet model, (3, 4 and 7 of the ISO model), see Figure 2. Their purpose is to control access to and from a protected network. Note that a firewall can be placed between any two networks, for example between a corporate business network and its R&D network. In general, a firewall is placed between a high security domain and a lower security domain.

A firewall system operating at layers 3 and 4 is sometimes called a packet filtering router or a screening router. Its purpose is to filter IP and ICMP packets and TCP/UDP ports. The router will have several ports and be able to route and filter the packets according to the filtering rules. Packet filters can also be built in software and run on dual homed PCs, but whilst these can filter packets they are not able to route them to different networks.

A firewall at layer 5 Internet (7 ISO) is sometimes called a bastion host, application gateway, proxy server or guardian system. Its purpose is to filter the service provided by the application.

It is also possible to operate a firewall system at Layer 2 (the link level) e.g. by configuring an Ethernet bridge to only forward certain packets, but this is not very common. The Inspection Module from Checkpoint's Firewall 1 product operates between the link and network layers and inspects packets before letting them pass through the firewall.

Packet Filtering Firewall

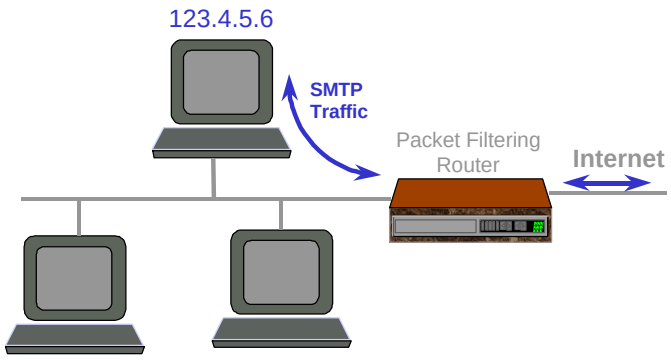

Figure 3

6. Packet Filtering Router

Packet filtering routers were the first type of firewall to be invented. A packet filtering router should be able to filter IP packets based on the following four fields:

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

Filtering is used to:

- block connections from specific hosts or networks
- block connections to specific hosts or networks
- block connections to specific ports
- block connections from specific ports

When configuring a router, it is usually possible to specify *all* ports or hosts, as well as specific ones. Packet filtering routers have fast performance, since the IP packets are either forwarded or dropped without inspecting their contents (other than the address and port fields). Packet filtering routers are equivalent to guards who ask someone "where are you from and where are you going to" and if the answer is OK, the person is let into the building.

For example, suppose an Internet access security policy stated that the only Internet access allowed was incoming and outgoing Email. Assuming that the organisation's Email server was located on host 123.4.5.6, then the router would be configured in the following way:

Type	**SourceAddr**	**DestAddr**	**SourcePort**	**DestPort**	**Action**
tcp	*	123.4.5.6	>1023	25	permit
tcp	123.4.5.6	*	>1023	25	permit
*	*	*	*	*	deny.

Note. * means any address.

Note. It is conventional for SMTP mail switches to always listen for incoming messages on port 25 (the well known port number), and to send messages on port numbers 1024 upwards.

The first rule allows incoming Email from any address to be sent to the Email server, the second rule allows outgoing Email to be sent from the Email server to any address, whereas the last rule forbids any other traffic from passing through the router.

7. Problems with Packet Filtering Routers

Packet filtering routers are a vital component of a firewall system, but they should only be considered as a first line of defence, since they do have a number of deficiencies.

1. They can be complex to configure (the rule set can be large, particularly when many services are supported), and there is no automatic way of checking the correctness of the rules i.e. that the rules correctly implement the security policy. Furthermore, if the router does not support logging of calls, there is no way of knowing if supposedly disallowed packets are actually getting through via a hole in the rules.
2. If some members of staff have special requirements for Internet access, then new rules may have to be added for their machines. This further complicates the rule set, maybe making it too complex to manage. Furthermore this access is at the wrong level of granularity, since the machine rather than the user is being given permission. Users are not authenticated, only the packets are checked.
3. Some basic routers do not allow TCP/UDP filtering, and this makes it impossible to implement certain security policies e.g. the one given in the example above.
4. You cannot filter between different ISO protocols running over TCP/IP. RFC 1006 specifies how ISO applications such as X.500 and X.400 may run over TCP/IP. However, all of the ISO applications must connect to port 102, on which the RFC 1006 service sits.
5. Finally, packet filtering routers are not very secure, since the contents of the packets are not inspected (only their headers) so anything can be being passed through e.g. viruses, unauthorised delete commands etc. Finally, the senders of the packets are not authenticated.

In order to overcome some of these deficiencies, more of the contents of the packets need to be inspected. This led to application level firewalls and more recently to the stateful packet inspection module from Checkpoint.

8. Stateful Packet Inspections

This is a software module that runs in the operating system of a Windows or Unix PC firewall, and inspects the packets that are arriving. The inspection is driven by security rules configured into the machine by the security officer. Headers from all seven layers of the ISO model are inspected, and information about the packets is fed into dynamic state tables that store information about the connection. The cumulative data in the tables is then used in evaluating subsequent packets on the same connection and subsequent connection attempts.

Whilst this technology is more secure than simple packet filtering routers, it is not as secure as application gateways, as the full application layer data is not inspected. However, it does perform faster than application proxies. Stateful inspection is similar to a security guard that asks who are you, where are you going, and what are you carrying, before he lets you into the building.

Note that this technology is patented by Checkpoint, the manufacturers of FireWall-1.

9. Application Level Firewalls

An application level firewall is created by installing a (bastion) host computer running the appropriate application(s), between the packet filtering router and the intranet. The packet filtering router directs all calls from the Internet to the application level firewall.

The application(s) running on the host are not usually full blown versions of the application(s), but rather are slimmed down proxy services that simply filter the messages at the application level, letting some messages through, rejecting other messages, and modifying others before accepting them.

If the host does not run a particular application proxy service, then calls to this application will not usually pass through the firewall to/from the Internet. In other words, all services **not** running on the firewall are blocked. Common application proxies, supported by most application firewalls suppliers are FTP, SMTP, HTTP and Telnet.

Application proxies are similar to a security guard who asks you why you want to enter the building and what are you carrying, and if he does not like your answer he will refuse you entry, or he may direct you to another person, or even remove some of your items or substitute them before letting you pass through. He may even take things off you before you can leave the building.

FTP poses a security threat because confidential information may be exported from the organisation, or bogus information may be deposited in the organisation's file store. The FTP proxy allows FTP commands to be selectively blocked according to source and destination addresses. For example, if the organisation has information that it wishes to publish on the Internet, the proxy would forbid sending *put* commands (i.e. writing) to the relevant FTP server and directory. If the organisation wishes customers to send files to it, then the FTP proxy can ensure that *dir* and *get* commands are blocked, and that the FTP connection is sent to the correct system and directory.

SMTP poses a security threat because mail servers (often the buggy *sendmail* program on UNIX systems) run with system level permissions in order to deliver incoming mail to users mailboxes. Hackers can initiate an interactive session with a mail server (by hand typing in commands or writing their own programs) and exploit its system level privileges. The SMTP proxy which runs on the firewall isolates the internal Email system from incoming Internet mail, thereby preventing Internet users from directly interfering with a mail server. Incoming mail is spooled in a reserved

directory on the firewall host, by the proxy SMTP mail program that runs without system privileges. The remote Email sender is then disconnected before any harm can be done. Another process picks up the mail from the reserved directory and forwards it to the internal Email system.

TELNET allows users to login to remote machines. This can be a security risk if remote users are allowed to login to the organisation's computers with standard username/password pairs, given the inherent weaknesses with password based systems. The Telnet proxy can be configured to state which systems can make calls to it, and which systems it will permit to be called. A typical configuration will be to allow internal users to call the Internet, but not vice versa.

HTTP accesses remote web pages. HTTP proxies can filter the various HTTP commands (methods) such as POST, PUT and DELETE as well as filter the URLs (e.g. forbid connections to .com sites)

In addition, all of the application proxies will provide logging of the incoming and outgoing sessions, and will authenticate the users. However, rather than each proxy having its own authentication service, it is beneficial if all proxies can make use of a common authentication module that runs on the firewall.

We also want to make sure that the data being transferred is virus free, therefore we need Content Filtering as well.

10. Content Filtering

With content filtering, the application data is handed over to a content filtering server that unpacks the data to see what is inside, and harmful content is then disposed of. For example, zipped files are unzipped first to see what is inside them. If the content contains a virus it will be discarded or disinfected. (Note, this requires that organisations regularly update their virus checking software, as new viruses are found daily.) File types are identified (not from the filename extension but from their content) and undesirable types e.g. executables can be removed, according to the security policy. Alternatively, if imported code is digitally signed, the author/signer can be checked to see if he is on a trusted list of signers and then the file can be accepted. Text files can be scanned for a list of undesirable key words (e.g. swear words or explicit sexual language). Finally, incoming http Java or ActiveX applets can be removed if this is company policy. Content filtering is like the security guard that empties your pockets, and gives you a full body check both on entering and leaving a building.

The biggest vendor of content checking software is Checkpoint with its MIMEsweeper family of products (that include MAILsweeper and WEBsweeper).

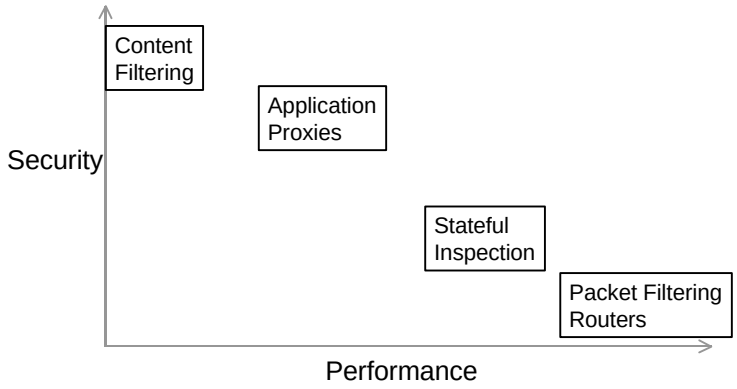

Figure 3

The biggest problem with scanning and filtering all the packet contents as they pass through the firewall, is the amount of processing time this takes, see Figure 3. Consequently, large servers are needed if all incoming data is to be screened.

11. Authentication

It has already been noted that simple passwords can not be relied upon to provide authentication information over the Internet. Something stronger is needed. The logical place to site the strong authentication functionality is in the firewall. An increasingly common authentication method is the use of **one-time passwords or hashed passwords**. But digital signatures are also becoming more popular as PKIs get implemented. Digital signatures rely on asymmetric encryption. The sender digitally signs a message, by appending to it a digital summary of the message (called a message digest), encrypted with his private key. The firewall can decipher the digital signature using the sender's public key. The firewall can also compute the message digest and compare this to the deciphered one. If both digests are the same, the message is authentic (it must have come from the owner of the private key and it has not been tampered with during transfer).

SOCKS authentication was one of the first general authentication mechanisms to be placed in a firewall, that allows remote applications to authenticate to the firewall. RADIUS is the Internet draft standard for dial in user authentication to a firewall.

11.1. SOCKS Authentication

SOCKS provides an authentication layer for the firewall that can be used by all application proxies. Calls come into the SOCKS service, are authenticated by it, then a call is opened up to the application proxy which does further application level filtering before making a call to the application on the intranet, see Figure 4.

SOCKS Authentication Service

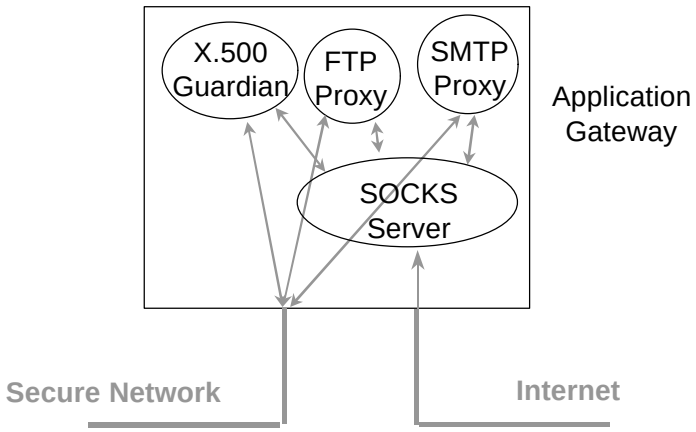

Figure 4

SOCKSv5 operates as follows. A TCP client opens up a connection to a SOCKS server at port 1080 in the firewall. The client negotiates an authentication method, then authenticates to the SOCKS server. If successful, the client sends a Relay Request to the SOCKS server. The SOCKS server then either relays the request to the requested server or rejects the request. If accepted, thereafter messages between the application server and the client are relayed via the SOCKS server. A full description of SOCKSv5 can be found in [1].

A disadvantage of SOCKSv5 is that it requires modified TCP software in the client system. Fortunately this is now widely implemented, and is supported for example in Netscape and Internet Explorer, plus freely available implementations of the SOCKS library and server are available for download from the Internet.

Authentication methods primarily supported by SOCKS are username password and GSS-API. But this is not such a wide range, and the password is sent in the clear so it is open to sniffing attacks.

11.2. RADIUS

The Remote Authentication Dial In User Service (RADIUS) is specified in RFC 2865 [2]. The mode of operation is as follows:

1. The user dials into the network via a modem. The network can be the corporate network running its own modems, or it could be an Internet Service Provider.
2. The receiving computer acts as a RADIUS client, and will usually ask the user for his username and password.
3. The RADIUS client sends an Access Request message to the RADIUS server including the username and password (which is encoded using MD5 to stop sniffing -see later).

Remote Authentication Dial In User Service (RADIUS)

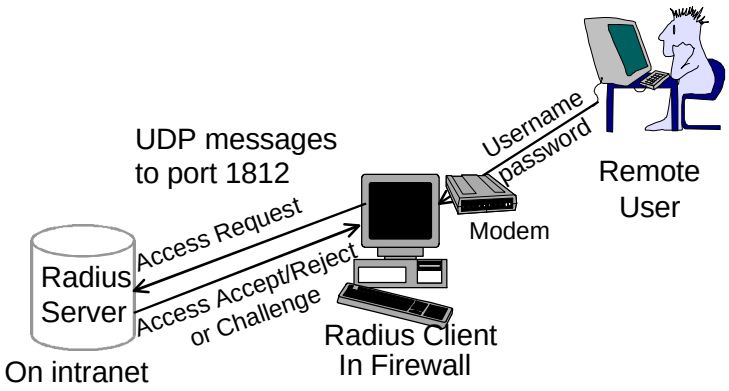

Figure 5

4. The RADIUS server authenticates the user using its local database and sends back an Access Accept or Access Reject message to the RADIUS client. If the message is an accept, it can include other information such as servers and ports the user is allowed to access.

Where Challenge/Response authentication is supported, the Access Request message will contain the user's name, and the RADIUS server will return an Access Challenge message containing a random number/string. This is sent to the user by the RADIUS client and the user must type in the correct reply. This is typically calculated using a one time password device (card or software). The RADIUS client then sends a second Access Request message containing the user's reply as the password. The server can determine if this is the correct reply and if so send an Access Accept message.

The RADIUS client and server share a secret (such as a long password) so that they can authenticate each other. When password based user authentication is being used, the Access Request message sent by the RADIUS client contains a 16 octet random number (called the authenticator), generated by the client. Then, in order to authenticate itself and the user to the server, it takes the authenticator and the shared secret, and hashes them using the MD5 algorithm. This produces a 16 octet (128 bit) number which is then XOR'ed with the user's password to produce another 16 octet number. This is carried in the Access Request message as the value of the password attribute (see Figure 6).

The authenticator is used in the Access Accept/Reject message so that the RADIUS client can authenticate the reply from the RADIUS server. Only this time the whole response message, the request authenticator and the shared secret are concatenated and hashed to form a 128 bit number which is sent as the response authenticator.

If the authentication method is challenge-response, then the first Access Accept message will not have a valid user's password (as it has not been computed yet).

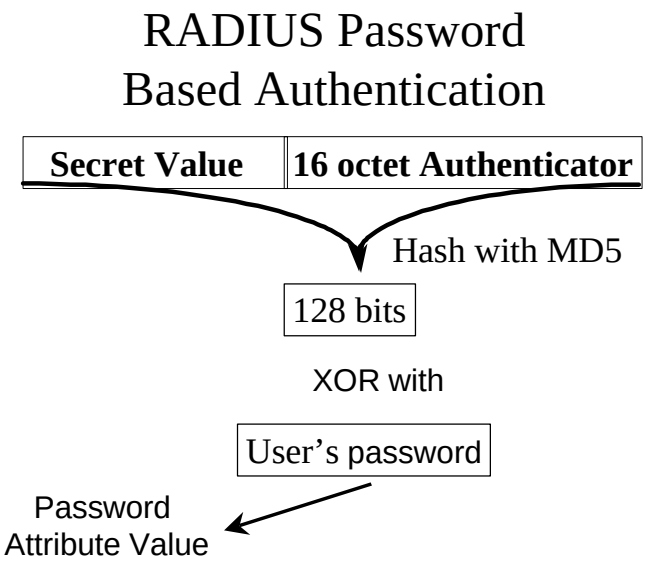

Figure 6

Therefore the password attribute can be blank or set to a fixed string such as "challenge required".

All messages are sent using UDP, and the RADIUS well known port number is 1812.

The advantages of RADIUS are that it is an open protocol, an Internet Draft Standard, supports a wide range of authentication mechanisms, is widely supported by vendors, and is extensible. RADIUS protocol messages contain a series of attributes (type, length, value tuples). These are standardised and registered with the ICANN (formerly IANA), and new attributes can be added as the Internet community agree on a need for them. Already defined attributes include: user's name, user's password, RADIUS client's IP address, call-back number (for modem's which call the user back at home) etc.

11.3. Hardware Based One Time Passwords

An increasingly common authentication method is the use of **one-time passwords**. There are two popular variants of one-time passwords, one is based on a challenge response mechanism, the other on synchronised clocks.

With the challenge response mechanism, the user logs into the firewall, and the firewall passes the user a challenge, usually in the form of a numeric string. The user responds to the challenge with a one-time password that is computed from the string by his hardware/software according to a pre-defined encryption algorithm that is also known to the firewall (see Figure 7). One such system (SecureNet from Digital Pathways) relies on the user having a one-time password card the size of a credit card that is capable of computing the passwords. The card has a digital display, and requires a PIN number to be entered before it can be used. Another system (S/key from Bellcore) relies on software in the remote user's PC to compute the password (see next section).

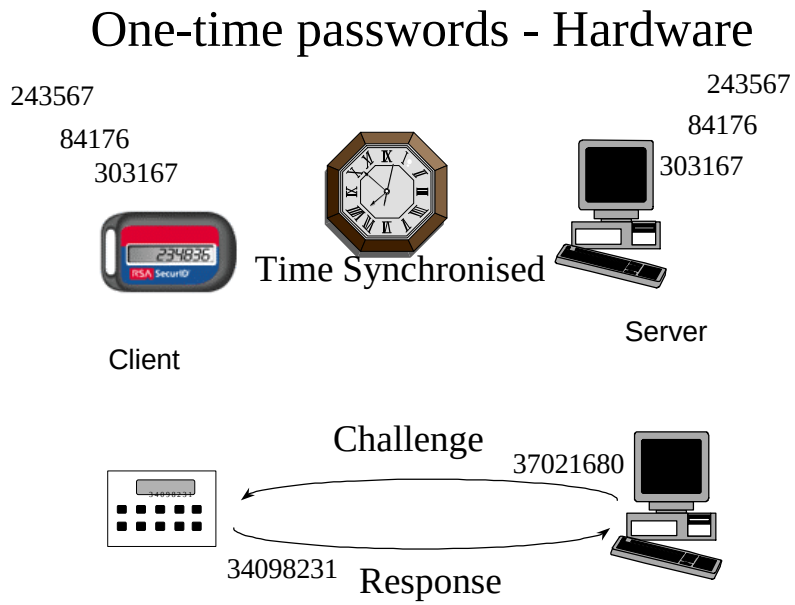

Figure 7

With the clock synchronised mechanism (e.g. SecureID from RSA Security), both the card and the firewall authentication system compute a new password every 60 seconds, according to a pre-defined encryption algorithm which uses the date and time, and a shared secret. This eliminates the need for a challenge string (see Figure 7). With the SecureID system, the user must transfer a PIN number plus the computed password, so that if the card is stolen it cannot be used by anyone else. This mechanism is sometimes referred to as Two Factor Authentication, as it is based on something I possess (the card) and something I know (the PIN).

11.4. Software Based One Time Passwords - S/Key

S/Key is a challenge-response one time password mechanism, and is widely supported by firewall vendors. Free S/Key implementations are available from the Internet. S/Key works as follows.

The server hashes up the user's password plus a random seed word a large number of times (say a thousand times) and stores the resulting 128 bit number. When the user asks to log in, the server returns a challenge comprising the seed word and the number 999 (one less than the n^{th} hash stored). The user is asked for his password, then his PC computes a hash of the password and seed word, then repeats this another 998 times and sends the resulting 128 bit hash to the server. This number is usually sent as ASCII words rather than binary, to stop the eight bit of each byte possibly being corrupted during transfer. The server takes the incoming number, hashes it once and compares it with its stored value. If they are the same it knows the user is authentic and allows the login. It then stores the hash it has been given. The next time the user wants to login, the server returns the seed word and the number 998, and the whole process is repeated. The user can login another 997 times until number 1 is reached. The server then has to invent a new random seed word and hash this with the password a thousand times and store it. The whole process then starts again.

Figure 8

The system works against sniffers, since it is not possible to compute the $(n-1)^{th}$ hash (you can never work out what the input to a hashing algorithm is). Therefore if an attacker captures the one time password this will not help him, as he cannot work out what the next one will be (he can only determine what the previous one was).

11.5. Virtual Private Networks

Virtual private networks connect two intranets, or an intranet and a remote user, together via an authenticated and encrypted link (see Figure 9). This secure tunnel makes the traffic confidential, authentic and un-replayable as it traverses the Internet. Most firewall vendors now provide VPN software as part of their firewall offerings.

The security is provided at the IP layer, by encapsulating each IP packet within an outer IP packet. The outer IP header directs the packet through the Internet to the remote firewall. Immediately after the outer IP header is an IPsec header that describes the security that has been applied to the encapsulated IP packet. (This is either an Authentication Header for packet authentication, anti-replay and integrity, or an Encapsulated Security Payload header for authentication, confidentiality, anti-replay and integrity.) The encapsulated packet contains the IP header for the final destination, followed by the payload (usually higher layer headers and higher layer data).

If the encapsulated data has been encrypted using symmetric encryption and a shared secret key, then this key has to be made available to both ends of the VPN. This can either be manually configured into both firewalls or it may be automatically generated by using the Internet Key Exchange protocol, which uses public key cryptography to distribute the key to the two firewalls.

Whilst IP Security (IPsec) [3] is optional in IPv4, it is mandatory in IPv6.

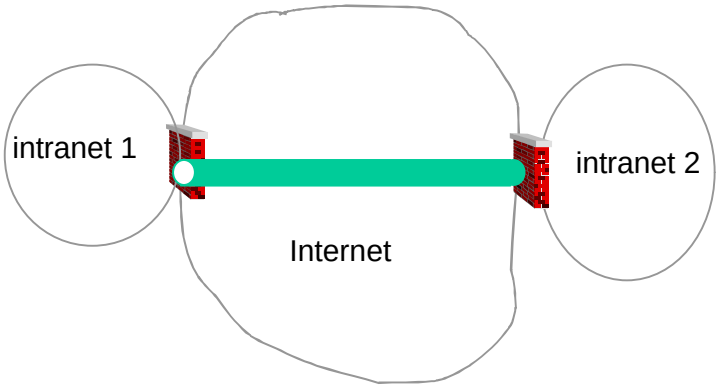

Figure 9

12. Firewall Configurations

12.1. The Dual Homed Gateway

This is a secure firewall design comprising an application gateway and a packet filtering router. It is called “dual homed” because the gateway has two network interfaces, one attached to the Internet, the other to the organisation's network. Only applications with proxy services on the application gateway are able to operate through the firewall. Since IP forwarding is disabled in the host, IP packets must be directed to one of the proxy servers on the host, or be rejected. Some manufacturers build the packet filtering capability and the application proxies into one box, thereby simplifying the design (but removing the possibility of having an optional info server and modems attached to the screened subnet, see Figure 10). The disadvantages of the dual homed gateway are that it may be a bottleneck to performance, and it may be too secure for some sites (!) since it is not possible to let trusted applications bypass the firewall and communicate directly with peers on the Internet. They must have a proxy service in the firewall.

12.2. The Screened Host Gateway

The screened host gateway is similar to the above, but more flexible and less secure, since trusted traffic may pass directly from the Internet into the private network, thereby bypassing the application gateway. In this design the application gateway only needs a single network connection (see Figure 11).

The IP router will normally be configured to pass Internet traffic to the application gateway or to reject it. Traffic from the corporate network to the Internet will also be rejected, unless it originates from the application gateway. The only exception to these rules will be for trusted traffic that will be allowed straight through.

Dual Homed Gateway Firewall

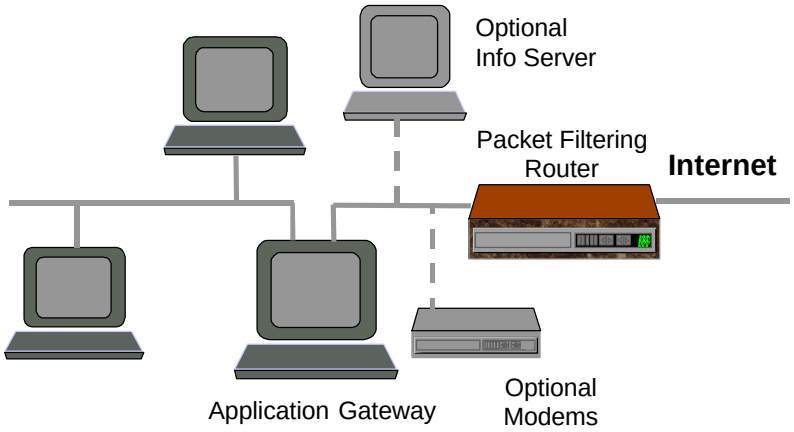

Figure 10

Screened Host Gateway

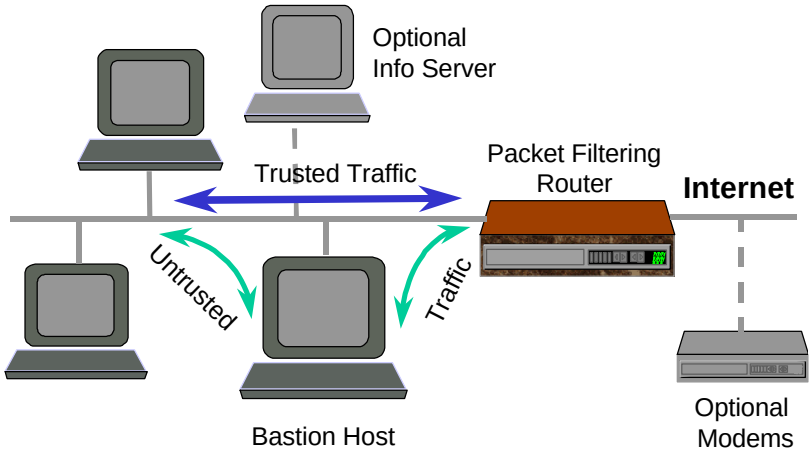

Figure 11

12.3. The Screened Subnet Gateway

This configuration creates a small isolated network between the Internet and the corporate network, which is sometimes referred to as the demilitarised zone (DMZ), see Figure 12. The advantages of this configuration is that multiple hosts and gateways can be stationed in the DMZ, thereby achieving a much greater throughput to the Internet than the other configurations; plus the configuration is very secure as two packet filtering routers are there to protect the corporate network.

Screened Subnet Gateway

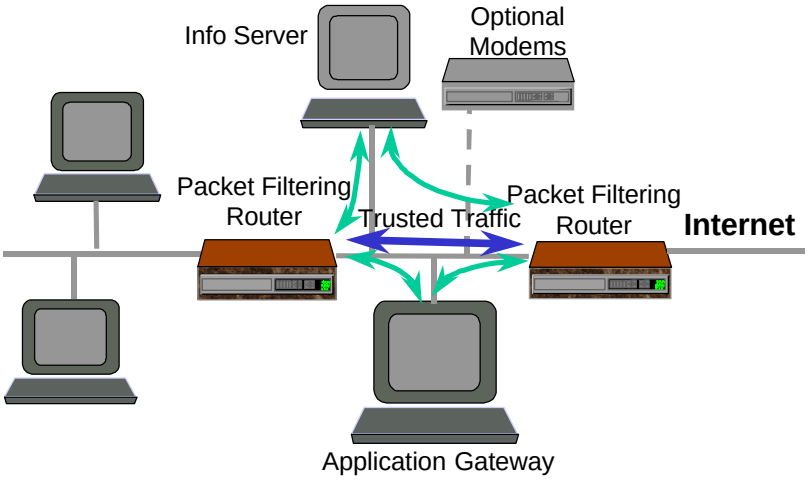

Figure 12

The IP router on the Internet side will only let through Internet traffic that is destined for a host in the DMZ (and vice versa). The IP router on the corporate network side will only let site traffic pass to a host in the DMZ (and vice versa).

This system is as secure as the dual homed gateway, but it is also possible to allow trusted traffic to pass straight through the DMZ if required. This configuration is of course more expensive to implement!

12.4. Double Proxying and a DMZ

Double Proxying and a DMZ

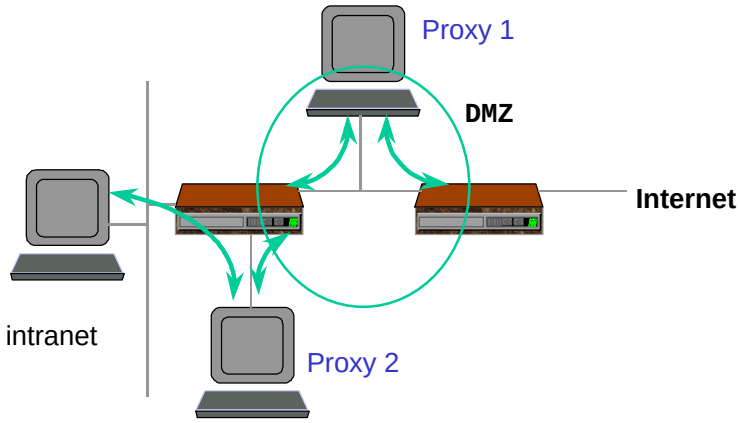

Figure 13

The configuration shown in Figure 13 is even more secure that the screened subnet seen in the previous section. It is used by a bank to protect its internal network from direct access from the Internet. Users from the Internet have to pass through two application proxies before they can access the bank's intranet.

This shows that there really is no limit to how complex a firewall configuration can be. The only limitations are the cost and performance implications of building ulta-secure firewall configurations.

References

[1] SOCKS Protocol Version 5. M. Leech, M. Ganis, Y. Lee, R. Kuris, D. Koblas & L. Jones. RFC 1928, April 1996. (Status: PROPOSED STANDARD).
[2] Remote Authentication Dial In User Service (RADIUS). C. Rigney, S. Willens, A. Rubens, W. Simpson. RFC 2865, June 2000. (Obsoletes RFC2138) (Status: DRAFT STANDARD)
[3] Security Architecture for the Internet Protocol. S. Kent, R. Atkinson. RFC 2401, November 1998.(Status: PROPOSED STANDARD)