ELSEVIER

Computer Communications 21 (1998) 1107–1123

computer communications

Tutorial

Internet/Intranet firewall security—policy, architecture and transaction services

Ray Hunt*

Department of Computer Science, University of Canterbury, Private Bag 4800, Christchurch, New Zealand

Received 22 September 1997; received in revised form 1 April 1998; accepted 3 April 1998

Abstract

The development of Internet/Intranet security is of paramount importance to organisations that plan to gain the economic benefits from interconnection with the Internet. This paper commences by examining firewall policy, focusing on both network service access policy and firewall design policy. Various firewall architectures, ranging from simple packet filters through to screened subnets and proxy gateways, are then discussed. Finally, the various mechanisms by which transactions can be secured over the Internet/Intranet are covered. These include encrypted tunnelling, IPv6, point-to-point tunnelling protocol, secure sockets layer, secure electronic transactions and secure multipart Internet mail encoding. © 1998 Elsevier Science B.V.

Keywords: Firewall design policy; Network service access policy; Packet filter/screening router; Dual-homed gateway; Screened host/subnet; Proxy gateway; Encrypted tunnelling; IPv6; PPTP; SSL; SET; S/MIME

1. Firewall policy

A firewall is a method of achieving security between trusted and untrusted networks and the choice, configuration and operation of a firewall is defined by the policy. The policy defines the services and type of access permitted between trusted and untrusted domains. Therefore, a firewall can be viewed as both a policy and the implementation of that policy in terms of network configuration, host systems, routers, encryption tunnels, authentication procedures and applications systems.

The definition of a firewall policy requires a clear explication of the security perimeter, since different firewall architectures provide different levels of guarantee against attack. Another important term is ''zone of risk'' that generally applies to TCP/IP capable networks, although networks using other protocols such as Netware/IPX, DECnet and SNA can also be vulnerable.

In principle, the zone of risk covers all networks and servers connected to the Internet, including the Internet backbone and related network infrastructures. The objective of the firewall policy is to minimise the organisation's zone of risk by removing the possibility of attack from an external network. In other words the firewall becomes the zone of risk for the trusted network.

It is widely accepted that there is a real risk from insider threats, and it is often stated that there are more insider attacks (for which a firewall is of little value) than external attacks [1]. Insiders usually have more direct access to the systems and the opportunity to abuse privileges. For example, many workstations can be easily reconfigured to grant privileged access and it is then a simple task to run a protocol analyser or decode software. Also, most standard TCP/IP applications, such as Telnet, FTP, rlogin, etc., have weak authentication control and passwords are transmitted in cleartext.

There are two levels of policy that influence the design, installation and use of a firewall:

- network service access policy (NSAP)
- firewall design policy (FDP).

1.1. NSAP

The NSAP defines which services are to be explicitly allowed or denied between trusted and untrusted networks, together with the way in which these services are to be used as well as any conditions for exception to this policy.

* Tel.: +64 3 3642347; fax: +64 3 3642569; e-mail: ray@cosc.canterbury.ac.nz

PII S0140-3664(98)00173-X

The NSAP should be an extension to existing business policy that will have already addressed the following issues:

- information value—what value does management places on information?
- responsibility—who is responsible for ensuring the protection of the organisation's information from untrusted networks?
- commitment—what is the organisation's commitment to protecting its information resources?
- domains—what domains should or should not be protected?

Further, business policy should already have implemented controls on such systems as

- virus scanning[1]
- physical security access
- floppy disk controls
- RAID back-up systems.

At the highest level the organisational policy might state:

- information is the strategic resource for the organisation;
- the availability, integrity, authenticity and confidentiality of the information will be protected by every cost-effective measure possible;
- ensuring the availability, integrity, authenticity and confidentiality of the information is a priority for all users at all levels of the company.

Below this level, specific policies are implemented which cover issues such as

- access to services (dial-in, dial-out)
- version controls
- user authentication
- trusted/untrusted network access.

It is at this level that the firewall's NSAP is formulated.

The NSAP must be drafted before the firewall is implemented. It must provide a balance between protecting the trusted network from known risks while providing users with convenient access to the untrusted network. Further, if a firewall denies access to certain services on an untrusted network, it is essential that the NSAP ensures that these controls are not circumvented or disabled. A typical NSAP might

- allow no access to applications or services on the trusted network from the Internet;
- as above, but allow access to a subset of applications or services by way of a secure server (e.g. bastion host);
- allow access from the Internet to selected applications on the trusted network (e.g. e-mail) in conjunction with strict authentication procedures (e.g. challenge/response and one-time password controls).

1.2. FDP

FDP defines how the firewall implements restricted access and service filtering specified by the NSAP and addresses issues such as

- IP address filtering
- encryption tunnelling
- secure socket control to facilitate application access
- audit and accounting control.

This policy is specific to the firewall and defines the rules and procedures necessary to implement the NSAP, but it must take account of the capabilities and limitations of the particular firewall platform as well as the threats and vulnerabilities associated with TCP/IP. For example, if the NSAP forbids access to all applications on the trusted network, then implementing a firewall by way of a packet filtering router is extremely risky.

In principle a firewall can

- permit any service unless it is specifically disallowed
- deny any service unless it is specifically permitted.

However, in practice, only the latter option is used. The first option might unintentionally allow denied services to run on non-standard TCP/UDP ports. Further, some services such as FTP, RPC and X-Windows are difficult to filter [2].

Depending upon the various security and flexibility requirements, some firewalls are more appropriate than others, which means that the NSAP must be carefully designed before the firewall is implemented. For example, dual-homed gateways (Section 2.2.1) and screened subnets (Section 2.2.3) can both be used to implement a ''deny all'' firewall. However, the dual-homed gateway is cheaper but also less flexible than the screened subnet.

In order to arrive at a successful design policy, together with a platform that implements this policy, it is usual to start by restricting all access from the untrusted to the trusted network, and then to specify the following [3].

- What Internet services will the organisation use (e.g. e-mail, Telnet, FTP, World Wide Web (WWW))?
- Where will these services be used from (intra-company, between branches, on a mobile or dial-in basis, by subsidiary organisations, etc.)?
- What additional security features will be needed (e.g. one-time password control, authentication procedures, encryption tunnels, secure sockets, point-to-point encryption, dial-in/dial-back procedures. etc.)?
- What risks result from the provision of these services? E.g. is a 40-bit RSA [4] encryption key adequate for certain government or banking applications? Is dial-in access without formal authentication procedures an acceptable risk?
- What is the cost (e.g. financial, inconvenience) of providing these services? For example, how is key distribution handled? What is the cost of managing a dedicated authentication server?

[1] Contrary to popular belief, firewalls can scan for viruses. This may require scanning at the application layer (mail or file headers). Most common antivirus products such as McAfee, F-Prot, Dr Solomon, Symantecs and Norton's AntiVirus can be configured to achieve firewall virus control.

- What is the balance between usability and security (e.g. if a particular service is too expensive or risky to use should its use be forbidden, thus creating great inconvenience)?

Some services that are inherently insecure may, with the addition of certain technologies, be secured to pose little or no risk. For example, a remote Telnet session can be very vulnerable to packet sniffing for passwords, and would pose a high risk when connecting a machine to a trusted network over an untrusted network such as the Internet. However, with the addition of encryption or strong authentication techniques this risk can be dramatically reduced.

Implementation of the firewall based upon these considerations requires careful use of risk analysis so that the calculated level of risk can be compared with that deemed to be acceptable according to overall company policy [5]. This may result in a change to the initial policy. For example, if the original NSAP denied all dial-in access, certain exceptions to this rule may need to be considered so as to achieve some overall organisational objective.

1.3. Sample policies

1.3.1. Remote access policy

As a specific example a "remote user advanced authentication policy" might address dial-in user access from the Internet as well as authorised users on travel or working from home. All such connections should use the advanced authentication service of the firewall to access systems at the site. Policy should reflect that remote users might not access systems through unauthorised modems placed behind the firewall, as it takes only one captured password or one uncontrolled modem line to enable a backdoor around the firewall.

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

Therefore, both dial-in and dial-out capabilities should be incorporated into the design of the firewall. Forcing outside users to go through the advanced authentication of the firewall should be strongly reflected in policy. Policy might also prohibit the use of unauthorised modems attached to host systems if the modem capability is offered through the firewall.

Since users could run point-to-point protocol (PPP) to create new network connections into a site protected by a firewall, it needs to be considered as part of the overall access policy. Such connections are potentially a backdoor around the firewall, and may be an even greater danger than a simple dial-in connection.

1.3.2. Information server policy

A site providing public access to an information server may wish to incorporate appropriate access controls into the firewall design and policy should reflect the idea that the security of the site will not be compromised in the provisioning of an information service. For example, a Web server that is intended to provide access for Internet users may not need to be behind the firewall at all, as the information provided by this server resides on that machine rather than being drawn from systems on the internal network. As long as the machine is regularly backed up it can operate unencumbered by a firewall and simply be restored if attacked.

It is useful to make a distinction between two fundamentally different types of traffic:

- information-server traffic (traffic concerned with retrieving information from an organisation's information server);
- business traffic, such as e-mail, file transfer, transaction services, etc.

The two types of traffic have their own risks and do not necessarily need to be mixed with each other. Screened subnet firewalls (Section 2.2.3) allow information servers to be located on a subnet and, therefore, to be isolated from other site systems. This reduces the chance that an information server could be compromised and then used to attack site systems.

1.4. Policy evolution

Two considerations drive the formation of an FDP with respect to Internet connections:

- the risk to the organisation's internal information and systems from external threats, e.g. denial of service attacks, IP spoofing, etc.;
- the risk of sensitive organisational information being disclosed as it is transmitted across the Internet, e.g. password file capture, information leakage attacks (e.g. finger), etc.

Once the FDP has been drafted, maintenance and review are important ongoing activities.

1.4.1. Maintenance of the FDP

Unlike many organisational policies, the FDP is not static and may need to change on a day-by-day basis depending upon new vulnerabilities which arise. For example JAVA was considered to be a great invention and the industry was assured by SUN that it was not a security risk. Therefore, as browsers evolved to become JAVA aware, JAVA code (applets) passed through firewalls. It is likely that JAVA never appeared in any company's firewall policy as it was probably considered to be part of the WWW. One large multinational decreed from the highest level that JAVA be disabled on all browsers, thus demonstrating the dynamic nature of an FDP. Other examples of policy maintenance include changes to a network's filtering rules as well as rule changes resulting from the introduction of new services.

1.4.2. Review of the FDP

It is most important that the FDP remains under constant review to ensure that the policy reflects the current state of play. As a result of FDP maintenance, the original policy can become unrepresentative of reality and this can introduce security holes. Examples include: change of the systems expert; wrong versions of software being loaded following a system crash. In many of these cases problems may not be detected until after a security breach has occurred.

1.5. Installing and operating a firewall

Once the decision is made to use firewall technology to implement an organisation's security policy, it is then necessary to install a cost-effective firewall that provides an appropriate level of protection. In general, a firewall should provide the following levels of protection:

- support and not impose a security policy;
- support a "deny all services except those specifically permitted" design policy (even if this policy is not initially implemented);
- accommodate new facilities and services should an organisation's security policy change;
- contain advanced authentication measures, such as encryption, challenge/response systems, and should contain the hooks for installing these facilities;
- employ filtering techniques to permit or deny services to specified hosts as needed;
- use flexible and user-friendly IP filtering and be able to filter on as many attributes as possible, including source and destination IP address, source and destination TCP/UDP port, protocol type, and inbound/outbound interfaces;
- use proxy services for applications such as FTP and Telnet, so that advanced authentication measures can be utilised at the firewall.

It will also assist if the firewall supports proxies for services such as NTP, NNTP, X-Windows, Finger, HTTP, and certain web browser software (Section 2.2.4). The firewall should also have the ability to centralise simple mail transfer protocol (SMTP) access, thus reducing direct SMTP connections between site and remote systems. This results in centralised handling of site e-mail.

The firewall should have the ability to concentrate and filter dial-in access as well as logging suspicious activity. If the firewall requires an operating system such as UNIX or Windows NT, then a secured version of the operating system should be part of the firewall, with other security tools as necessary to ensure firewall host integrity. The operating system should have all patches installed and be developed in a manner that its strength and correctness is verifiable. It should be simple in design so that it can be understood and maintained.

Some organisations have the capability to put together their own firewalls, using available software components and equipment or by writing a firewall from scratch. Trusted Information Systems (TIS) Internet Firewall Toolkit [6] is a good example of a company that offers "firewall construction kits". At the same time, there are many vendors[2] [7] offering a wide range of services in firewall technology which include

- provision of the necessary hardware and software
- development of security policy and carrying out risk assessments
- security reviews and security training.

Consideration of the following questions may help an organisation decide whether or not it has the resources to install and operate a successful firewall.

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

As a general rule it is desirable that sites:

- standardise operating system versions and software to make installation of patches and security fixes more manageable;
- institute a program for efficient, site-wide installation of patches and new software;
- use services to assist in centralising system administration, if this will result in better administration and better security;
- perform periodic scans and checks of host systems to detect common vulnerabilities and errors in configuration.

2. Firewall architecture

There are many different interpretations of the term firewall and this can be a source of confusion. One basis for defining a firewall is the OSI 7-layer model (Fig. 1) which provides a clearer picture than does the TCP/IP model.

[2] Examples include Digital's Alta Vista Firewall, Secure Computing's Firewall for NT, Eagle NMS (Raptor Systems), ANS Interlock Service 3.06 (ANS CO + RE Systems), Borderware Firewall Server, Firewall-1 v2.0 (Checkpoint Software), Black Hole 3.0 (Milkyway Networks Corp.), Sidewinder, Gauntlet (Data General), GFX Internet Firewall (Global Technology Associates).

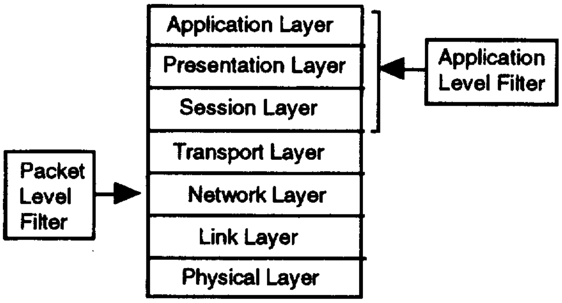

Fig. 1. Firewall architecture in relation to the OSI model.

It can be seen that firewall architecture consists of two levels:

- Packet-level firewalls that operate at the network (IP) and transport (TCP) layers. These are commonly referred to as screening routers or packet filters and block transmission of certain classes of traffic;
- application-level firewalls which operate at the session, presentation and application layers. They are usually implemented using dedicated hosts running specialised software and can also be referred to as bastion hosts or proxy servers, usually running under UNIX or Windows NT. They can also provide relay services to compensate for the effects of the filter(s).

Another important term often used in conjunction with a firewall is ''gateway'', and Internet firewalls are often referred to as secure Internet gateways. However, there is a more specific use of this term, as can be seen in Fig. 2. The network occupied by the gateway is often referred to as the demilitarised zone (DMZ) [8].

The gateway in the DMZ may consist of both an internal and external machine, as shown in Fig. 3. Normally these two gateways will have more open communication through the inside packet filter than the outside gateway has to other internal hosts. The outside packet filter can be used to protect the gateway from attack, while the inside gateway can be used to guard against the consequences of a compromised gateway.

2.1. Packet filters/screening routers

As packets pass through the router their filtering is based upon a set of rules established by the NSAP. Filtering based upon one of more of the following criteria are commonly applied:

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

Not all packet filters can filter on TCP/IP port numbers. Some can examine which of the network interfaces a packet arrived at and then use this as further filtering criterion.

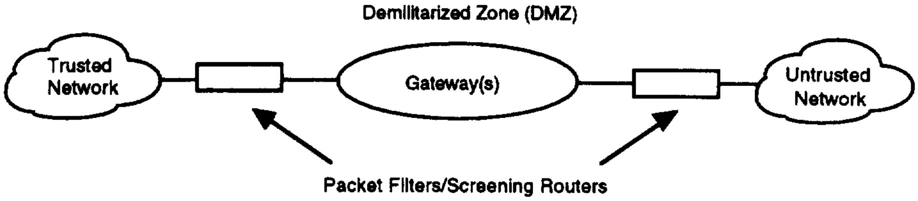

Fig. 2. Firewall design (filter/router and gateway).

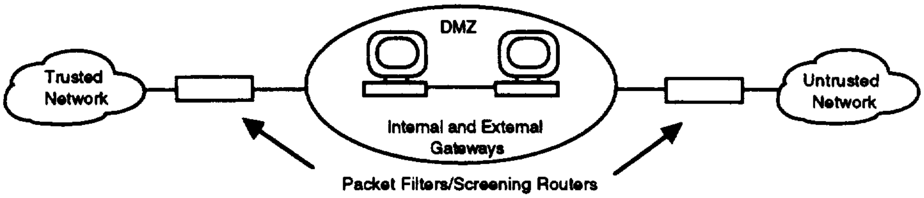
DMZ
Trusted Network
Internal and External Gateways
Untrusted Network
Packet Filters/Screening Routers

Fig. 3. Firewall design (filter/router with internal and external gateways).

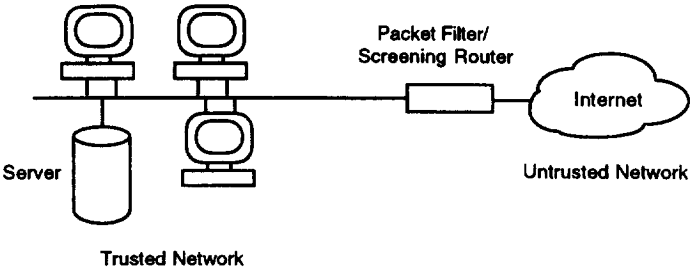

Fig. 4. Firewall using a packet filter/screening router.

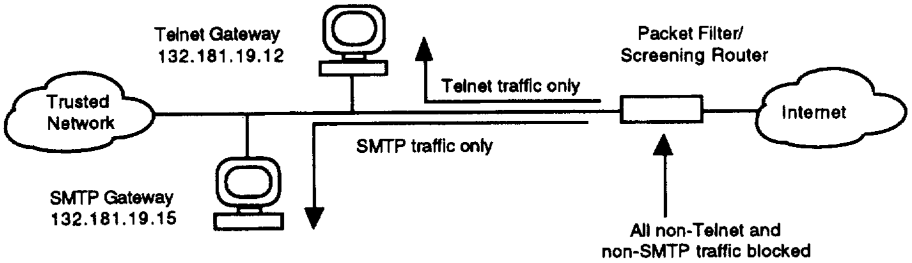

Fig. 5. Packet filtering on Telnet and SMTP.

Filtering can be used to block connections to or from specific hosts or networks (Fig. 4), as well as to block connections to specific ports. A site might wish to block connections from certain addresses, such as from hosts or sites that it considers being untrusted. Alternatively, a site may wish to block connections from all addresses external to the site (with certain exceptions, such as SMTP for receiving e-mail).

Adding TCP or UDP port filtering to IP address filtering results in a great deal of flexibility. Servers such as the Telnet daemon usually reside at specific ports (e.g. port 23). If a packet filter can block TCP or UDP connections to or from specific ports, then it is possible to implement policies that call for certain types of connection to be made to specific hosts, but not others. For example, a site may wish to block all incoming connections to all hosts except for several firewalls-related systems. At those systems, the site may wish to allow only specific services, such as SMTP for one system and Telnet or FTP connections to another system. By filtering on TCP or UDP ports, this policy can be implemented in a simple manner using a packet filtering router or a host with packet filtering capability.

Fig. 5 shows an example of packet filtering with a policy that permits only certain connections to a network of address 132.181.*.*:

- Telnet connections will only be allowed to the Telnet application machine (Telnet gateway) with an IP address of 132.181.19.12
- SMTP connections will only be allowed to the SMTP application machine (e-mail gateway) with an IP address of 132.181.19.15.

All other services and packets are to be blocked. This is a very basic example of packet filtering, and actual rules can become very complicated in practice.

2.1.1. Policing protocols

It is the NSAP that determines which protocols and fields are filtered, i.e. which systems should have Internet access and the type of access to permit. The following services are inherently vulnerable to abuse and are usually blocked at the firewall [3]:

- TFTP (port 69), trivial FTP, used for booting diskless workstations, terminal servers and routers, can also be used to read any file on the system if set up incorrectly;
- X-Windows (ports 6000 +) and OpenWindows (port 2000), can leak information from X-Window displays, including all keystrokes (intruders can even gain control of a server through the X-server);
- RPC (port 111), remote procedure call services including NIS and NFS, which can be used to steal system information such as passwords and read and write to files;
- rlogin, rsh, and rexec (ports 513, 514, and 512) services which, if improperly configured, can permit unauthorised access to accounts and commands.

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

- Telnet (port 23), often restricted to certain systems;
- FTP (ports 20 and 21), like Telnet (port 23), often restricted to certain systems;
- SMTP (port 25), often restricted to a central e-mail server;
- RIP (port 520), routing information protocol, can be spoofed to redirect packet routing;
- DNS (port 53) domain names service, contains names of hosts and information about hosts that could be helpful to attackers and it can be spoofed;
- UUCP (port 540) UNIX-to-UNIX CoPy, if improperly configured can be used for unauthorised access;
- NNTP (port 119) network news transfer protocol, used for reading network news;
- NTP (port 123) network time protocol, used for setting system clocks;
- gopher (port 70) and HTTP (port 80) information servers and client programs for gopher and WWW clients, should be restricted to an application gateway that contains proxy services.

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

Forged or spoofed packets can cause a real threat, as indicated above in relation to RIP and DNS. IP spoofing can be used to make a host look as though it is a trusted host by changing the IP number for example. Also, if source routing is used, the IP header contains the route that the packet will take to reach its destination and thus override routing table instructions. However, attackers who are attempting to circumvent security can generate source-routed packets with Telnet clients to ensure packets follow specific paths. Worse still, reply packets are intended to use the inverse of the original route and this can now permit a two-way connection which was supposed to be blocked.

2.1.2. Non-IP protocols

Most policing protocols outlined in Section 2.1.1 focus on IP. However, other protocols at the same level as IP—such as IPX, NetBeui and AppleTalk—have different packet header formats and filtering characteristics. These non-IPs are rarely used between organisations across the Internet. Some packet filters offer limited filtering support for non-IPs, but they are usually considerably less flexible than their IP equivalent. Many packet filters are configured to just drop non-IP packets. Most non-IPs can be handled by encapsulating them within IP packets. Packets filters then usually either permit or deny encapsulated protocols in their entirety [9].

Another reason to disable non-IP packets is because their escape from LAN domains can reveal important information, such as the LAN's primary architecture. Many of the older protocols, such as RIPX, NetBios and Digital's LAVC (VAX cluster keep alive packets), broadcast the presence of devices that support these protocols across large parts of the network and misconfigured routers may not limit this dangerous traffic to a single LAN segment.

2.1.3. Stateful inspection

Stateful inspection, sometimes also known as dynamic packet filtering, is a recent development by some firewall manufacturers and represents yet another approach towards firewalling. Stateful inspection intercepts packets at the network layer and examines these packets based on their communication state. This mandates the storing of state information from one or more packets as well as buffering and reassembling the datagram before an access decision can be made.

The advantage of this approach lies in its diversity and ability to support a variety of protocols and services. By not relying on the native stack of the firewall host for processing, as well as its ability to look back at past information related to the session, stateful inspection can apply any rule based on the communication content itself. Stateful inspection is usually implemented with support of the entire TCP/IP suite.

2.2. Application-level firewalls

Unfortunately, packet filtering routers have limitations and are frequently difficult to configure and update. Packet filtering rules are complex to specify and usually no testing facility exists for verifying the correctness of the rules (other than by exhaustive testing by hand). Some routers do not provide any audit capability, so that if a router's rules still let through dangerous packets, this may remain undetected until a break-in has occurred.

Exceptions to rules will often need to be made to allow certain types of access that normally would be blocked. However, exceptions to packet filtering rules can make the filtering rules so complex as to be unmanageable. For example, it is relatively straightforward to specify a rule to block all inbound connections to port 23 (the Telnet service). If exceptions are made and certain systems need to accept Telnet connections directly, then a rule for each system may need to be added (some packet filtering systems allow the sequential order of the filter rules to be significant). Sometimes the addition of certain rules can complicate the entire filtering scheme and open up further holes.

Advantages of application-level firewalls or gateways include:

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

Application firewalls or gateways can make some set-up and operational procedures more complex. For example, Telnet requires ports for both inbound and outbound traffic and the Telnet user has to remember to connect to the firewall and not the destination host. Sometimes client software has to be reconfigured to operate through the firewall, e.g. the set-up of Netscape requires proxy addresses to be allocated for common services such as HTTP and FTP.

Application firewalls also act as important filters for other protocols such as SMTP, FTP and X-Windows. For example, some application firewalls provide the capability to deny FTP get and put commands which prevents the uploading of a file to an anonymous FTP server and thus provides a higher degree of assurance than just relying on correct file setting permissions at the anonymous FTP server.

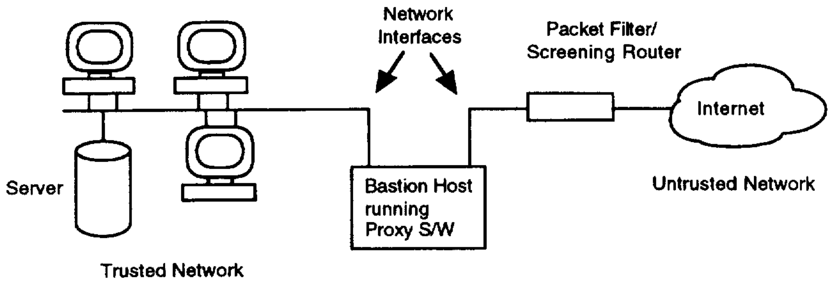

Fig. 6. Dual-homed gateway with two network interfaces.

Some Internet firewalls use a combination of a packet-filter screening computer or a hardware-router for controlling the lower layers of communication, and application gateways for the enabled applications. But even this combination may provide only limited transparency, flexibility and connectivity. It may also be expensive in terms of set-up, management, and expertise.

Another approach is for the firewall to employ an inspection module, applicable to a range of protocols. This can provide an interpretation of encapsulated data destined for higher layers. This has the potential to provide context-sensitive security for complex applications, which may be more effective than technologies which only have data in some of the layers available to them. For example, while application gateways have access only to the application layer and routers have access only to the lower layers, this inspection approach integrates the information gathered from all layers into a single inspection point.

In general, packet filters/screening routers and application gateways can form a powerful combination. Packets filtering via the router controls the lower architectural layers whereas application gateways offer fine-tuned controls appropriate for the applications.

2.2.1. Dual-homed gateway

The dual-homed gateway is an application-level firewall implemented without the use of a screening router. It has two network connection ports connected to each of the internal and external networks (Fig. 6). With IP forwarding disabled, a complete block of traffic between the two networks is possible.

There are two ways a user can access the external network from the internal network. The first is by direct logon to the gateway. This is not advisable as it makes this gateway directly vulnerable to password cracking, and can provide access directly to the firewall through software vulnerabilities, such as compiler bugs at the host. The biggest threat to the security of a dual-homed gateway arises when an attacker gains login access. Therefore, login should always be through application proxies on the dual home gateway.

The second and safest way for a connection to be made is through the application layer using a proxy service. A proxy service is an application which routes IP traffic from one port to another. Such an application can provide user authentication, auditing, and logging facilities. These features are a great improvement over packet filters/screening routers, which generally provide no more than rudimentary facilities.

Proxy software has to be written for each service, although basic proxy servers for standard TCP/IP services, such as Telnet, FTP, WWW, etc., are generally available for the Unix and Windows NT environment.

Once an attacker obtains login access to a dual-homed gateway the internal network is subject to intrusions. The zone of risk has been extended from the dual-homed host, to include the entire internal network.

The following is a list of sources from which an attack could be mounted [10]:

- weak permissions on the file system;
- internal network NFS-mounted volumes;
- permissions granted to Berkley r-utilities (e.g. rlogin, rcp (remote copy program), rdist (remote distribution)) through host equivalent files, such as .rhosts (list of available hosts), in users' home directories from user accounts that have been compromised;
- network backup programs that could restore excessive permissions;

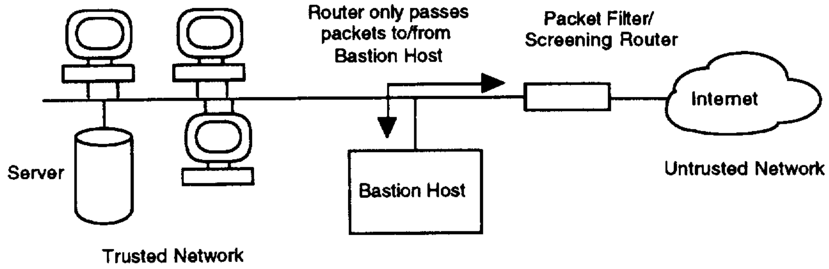
Router only passes packets to/from Bastion Host
Packet Filter/ Screening Router
Internet
Untrusted Network
Server
Bastion Host
Trusted Network

Fig. 7. Screened host gateway.

- use of administrative shell scripts that have not been properly secured;
- learning about the system from older software revision levels and release notes that have not been properly secured;
- installing older operating system kernels that have IP forwarding enabled, or installing an older version of the operating system kernel.

The key for the attacker is to gain enough system privileges to be able to change the Unix kernel variable ''ipforwarding'', which controls IP forwarding. Once this variable has been enabled the firewall has been completely subverted. Thus, the zone of risk is increased from including only the dual-homed gateway, to including all areas of the network.

Apart from disabling IP forwarding, there are a number of items that can be checked to ensure the security of a dual-homed gateway:

- remove all programming tools, including compilers, linkers, utilities, and services not specifically required for the operation of the dual-homed gateway;
- ensure programs that have SUID (Set User ID) and SGID (Set Group ID) permissions are removed if not required. Check that no excessive permissions on files and programs exist;
- use disk partitions so that denial-of-service attacks designed to fill all available disk space on a partition are confined to that partition only;
- remove unneeded system and special accounts, e.g. disable guest accounts, and maintenance accounts found on some proprietary systems (including screening routers);
- delete network services that are not required.

2.2.2. Screened host gateway

The screened host gateway is implemented using a packet filter and a bastion host. It is one of the most popular firewall architectures. The bastion host is usually placed on the internal network, with the packet filter configured such that the bastion host is the only machine reachable from the Internet (Fig. 7). To restrict Internet access further, the packet filter is generally configured to block all traffic not destined to specifically authorised ports on the bastion host. This has the effect of controlling the number of available services.

Major benefits of screened host gateways include reduction of router programming complexity, and improved connectivity for local users. As all traffic is passed through the bastion host, the rules for configuring the router table need only consider the bastion host's IP address. All other packets arriving at the inbound or outbound ports of the packet filter can be discarded, which greatly simplifies packet filter rules.

The zone of risk incorporates only the packet filter and bastion host. The security of this firewall architecture is determined by the accuracy of the packet filter rules in relation to the security policy, and the level of assurance regarding the software running on the bastion host. If an attacker gains entry to the bastion host then the threats to the internal network are similar to those of the dual-homed gateway.

There is, however, a major problem with this architecture which relates to the positioning of the bastion host on the internal network, and relying on the packet filter to control the traffic flow to and from it. If the packet filter is compromised, either through mis-configuration or through an attacker gaining access—such as via a proprietary maintenance account—then the entire internal network is at risk. Compromising the packet filter effectively subverts the bastion host and, once an attacker has control of the packet filter, all traffic can be routed to the external network.

A more secure implementation is to use a packet filter connected to a dual-homed gateway. This architecture ensures that the bastion host is not circumvented in the event of the packet filter being compromised. The attacker has to overcome the dual-homed gateway before the internal network is at risk. Of course, this architecture offers no improvement if an attacker is able to enter through the packet filter and compromises the bastion host directly.

2.2.3. Screened subnet

A screened subnet firewall architecture, consists of an isolated network positioned between the external and internal networks. This configuration allows non-critical hosts, such as WWW servers and anonymous FTP sites, to be placed outside the internal network. Bastion hosts are also

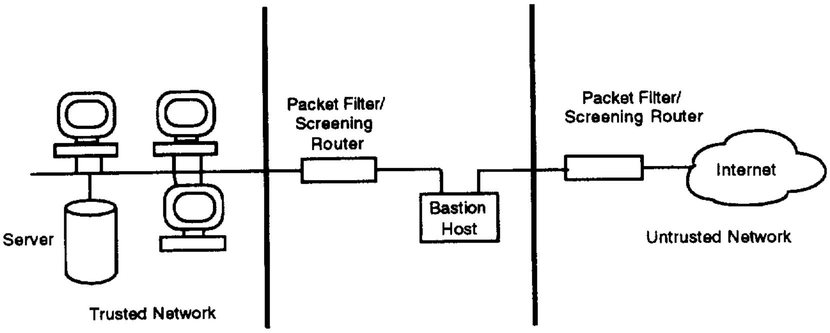
Packet Filter/ Screening Router
Packet Filter/ Screening Router
Internet
Bastion Host
Server
Untrusted Network
Trusted Network

Fig. 8. Screened subnet employing dual-homed gateway.

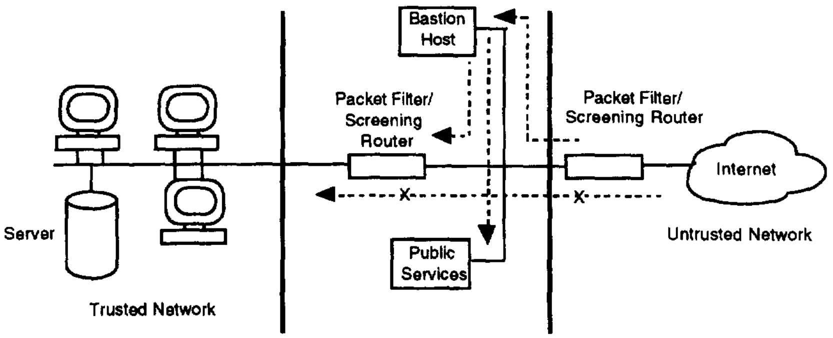

Fig. 9. Screened subnet employing bastion host and public services server.

placed on the subnet to provide interactive terminal sessions, or application-level firewalls. The screened subnet is generally considered to be the most secure firewall architecture. Two possible screened subnet scenarios are shown in Figs. 8 and 9.

A dual-homed gateway or bastion host provides the sole point of access to machines on the internal network, and forces all services through the firewall to be provided by application gateways. Also providing protection are two packet filters, one between the external network and subnet (known as the external router), the other between the subnet and internal network (known as the internal router). Therefore, the zone of risk for this configuration consists of only the two routers, and the dual-homed gateway or bastion host, as well as any other hosts placed on the subnet.

The strength of this firewall architecture comes from the fact that an attacker must subvert the external router, followed by the dual-homed gateway or bastion host, and finally the internal router. If the packet filters are configured so they cannot be managed remotely from the network, then subverting this firewall architecture without setting off alarms and appearing in audit logs would be very difficult.

As with the screened host gateway, if the packet filters can be compromised by logging in and reconfiguring their routing tables, the bastion host can be negated and the internal network put at risk. A drawback with this configuration is the extra level of complexity added to the definition of packet filter rules, especially if there are additional public hosts on the subnet.

2.2.4. Proxy gateway

A proxy server is a specific type of application firewall or gateway designed to mediate traffic between a protected network and the Internet. This application is called a proxy service, and the host running the proxy service is often referred to as an application gateway. Proxies are often used instead of router-based traffic controls to prevent traffic from passing directly between trusted networks and the Internet.

Users on an untrusted network, such as the Internet, can only see the proxy server and are therefore shielded from the trusted network's services and applications. The proxy allows only those services through for which a proxy has been specifically specified. For example if the application gateway specifies proxies for FTP and Telnet, then only those services will be allowed into the trusted network and all other services are disabled.

Many proxies contain additional auditing facilities to support user authentication and protocol specific security, for example an FTP proxy might be configured to permit incoming but block outgoing FTP. For some sites, this degree of security is important, as it guarantees that only those services that are deemed trustworthy are allowed through the firewall. It also prevents other untrusted services from being implemented behind the backs of the firewall administrators.

Proxy servers are application specific and software must be installed for each protocol for which a proxy is required. One popular set of proxy servers is the TIS Internet Firewall Toolkit that includes proxies for protocols such as Telnet, rlogin, FTP, X-Windows, HTTP/Web, and NNTP/Usenet news.

SOCKS is a generic proxy system that can be compiled into a client application to make it operate through a firewall. It consists of a set of near-compatible replacements for various system calls such as: socket, connect, bind, etc. Converting an application simply requires the replacement of the standard calls with the SOCKS versions and existing applications can run unchanged [11]. Its advantage is that it is easy to use, but it is not portable and is therefore unlikely to support additional authentication hooks or protocol-specific logging.

As an example, consider a site that blocks all incoming Telnet and FTP connections using a packet filtering router. The router allows Telnet and FTP packets to go to one host only—the Telnet/FTP proxy gateway. A user who wishes to connect inbound to a site system would have to connect first to the proxy gateway, and then to the destination host in the following sequence (Fig. 10):

- user first Telnets to the proxy gateway and enters the name of an internal host;
- the proxy checks the user's source IP address—probably via some internal host—and accepts or rejects it according to the access policy;
- the user may need to be authenticated (possibly using a challenge/response device);

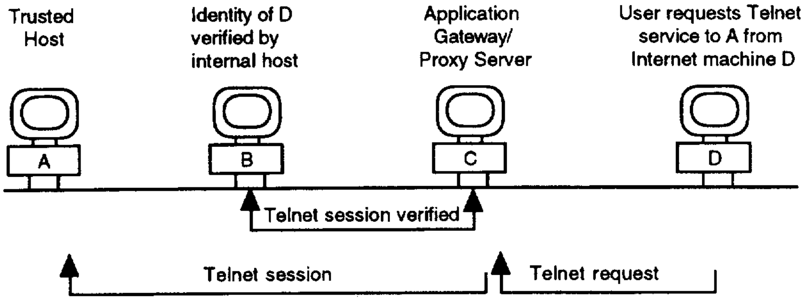
Trusted Host
Identity of D verified by internal host
Application Gateway/ Proxy Server
User requests Telnet service to A from Internet machine D
A
B
C
D
Telnet session verified
Telnet session
Telnet request

Fig. 10. Proxy used to log and filter Telnet session between Internet and trusted host.

- the proxy service creates a Telnet connection to the internal host;
- the proxy service then passes data between the two connections;
- the proxy gateway audits the connection.

2.2.5. Circuit-level gateway

Circuit-level gateways are similar to application-level gateways, with the distinction that instead of connections being mediated by the firewall, a virtual circuit is created between the external and internal hosts, which results in creating a hole in the firewall. Generally, circuit-level gateways are used to relay TCP connections from internal to external hosts. A certain amount of trust must be placed in the host opening the circuit, as the level of control the firewall has over the connection once it is established is less than that of the application-level gateway. For example, it is possible to piggyback restricted protocols over those already permitted.

3. Secure transactions on the Internet/Intranet

3.1. Security challenges and solutions

There are many challenges in building a full-service Intranet that provides safe communications and collaboration. As the exponential growth of the public Internet demonstrates, TCP/IP solves many problems in a remarkable way. However, TCP/IP was not designed to offer secure communication services. Because TCP/IP was not designed with security in mind, additional technology and policies must be brought to bear to solve, for example, the following typical security problems.

- How can users be authenticated to make sure they are who they claim to be? Standard Web protocols such as TCP/IP and HTTP make impersonating a person or an organisation relatively simple.
- How can authentication be performed without sending usernames and passwords across the network in cleartext?
- How can a single-user login service be provided to avoid costly account maintenance for all the servers (Web, proxy, directory, mail, news, etc.) across the enterprise?
- How is it possible ensure that these services not only work on an Intranet but also scale to the Internet? In other words, how is it possible to avoid managing a separate security scheme inside the firewall and a completely different scheme outside the firewall?
- How can privacy of communications be enforced for both real-time (e.g. data flowing between a Web client and server) and those with store-and-forward applications such as e-mail?
- How is it possible to ensure that messages have not been tampered with between the sender and the recipient?
- How can confidential documents be safeguarded to ensure that only authorised individuals have access to them?

There are a number of technologies available which provide the foundation for a wide variety of security services, including encryption, message integrity verification, authentication, and digital signatures, all of which are based upon cryptography. A variety of authentication and encryption techniques have become available in recent times to provide secure Intranet operation out of basic Internet services.

These techniques include encrypted tunnelling, IP next generation (IPng) and now called IPv6, point-to-point tunnelling protocol (PPTP) and layer 2 forwarding (L2F), secure sockets layer (SSL), secure electronic transactions (SET), secure multipart Internet mail encoding (S/MIME) which can be seen in the protocol stack in Fig. 11. The following sections provide a brief overview of how secure Internet transactions can be achieved.

3.2. Encrypted tunnelling

Tunnelling is a technique for encapsulating data for transparent transmission across an interconnecting network. For example, IPX traffic can be encapsulated in TCP/IP for transport across a network that does not recognise the IPX protocol. Such a link is transparent to the interconnected applications.

Encrypted tunnels take standard IP packets, encrypt and encapsulate them in TCP/IP packets for transport. Source and destination IP-based applications operate as normal, but the data between the two tunnel servers is encrypted.

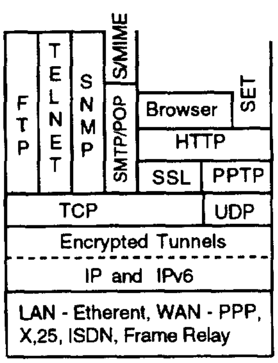

Fig. 11. Protocol architecture for secure Internet transaction services.

Frequently, users initiate connections to the Internet via the firewall, but considerable restrictions usually occur in the reverse direction. When setting up an Intranet using encrypted tunnels, the firewall can be configured to permit encrypted and authenticated data from Internet tunnel clients to pass through into the trusted network. The tunnelled data remains encrypted past the firewall inside the trusted network until it reaches the tunnel server, which may be on the same LAN as the firewall—or in fact anywhere in the trusted network.

Some vendors provide host-to-host or LAN-to-LAN encryption tunnels, whereas others permit the tunnel to terminate on the workstation. For example, Digital's Internet Personal Tunnel [12] makes use of client software running on a Windows 97/NT workstation. In contrast, the tunnel server can handle multiple workstation tunnel clients simultaneously. Authentication keys are stored at the workstation in encrypted form and are accessible by the use of a PIN. Different keys are used for each tunnel set up between the tunnel client and server.

Mobile, home or office-based PC users can connect to an Internet service provider (ISP) and establish a secure tunnel to the trusted network. Authorisation is based upon a PIN or challenge/response system (and not an IP address) followed by an encryption key pair used to authenticate the PC with the trusted tunnel server. This allows a user to roam the network, without being tied to a specific hardware address. Once an encryption tunnel has been set up to the trusted network, the PC can communicate anywhere within this network or be restricted by specific host policies. No special software is required within the trusted network since normal routing is used once the packets leave the tunnel and are decrypted. Neither is any special software required at the ISP, even though all traffic passing through is encrypted as can be seen in Fig. 12.

3.2.1. *Operation of encrypted tunnels*

When a tunnel client (group or personal) wishes to initiate a connection with an Internet Group Tunnel server, a connection request is sent over the network. The request may actually be addressed to a firewall, which relays the connection to the proper Internet Group Tunnel server. The connection request message contains an identification message that is encrypted by the client with the server's public key, and then decrypted by the server with its own private key.

The server's database contains a list of clients that are authorised to build tunnels. If the request is to be granted, the tunnel server sends an encrypted response using the client's public key, which is then decrypted by the client using its own private key. After this authentication sequence, the two parties exchange portions of a session key, which are combined to form a secret session key (Fig. 13). Session keys are periodically changed to enhance security. Each tunnel pair uses a public/private RSA key pair to authenticate each other. Once the tunnel is established, secret RC4 session keys are exchanged for use in encrypting user data. Within the USA a 128-bit key is used, whereas for international operation a 40-bit key is used.

The tunnel client is issued an IP address that is valid for the network of the Group Tunnel server. The tunnel client in effect has two addresses. One is appropriate for the network the client is directly connected to, and the other is appropriate for the network that has been tunnelled into. This will prevent a backdoor option for other workstations on the ISP's LAN from establishing a connection with the Intranet.

The clients routing tables are set up so that all communication intended for hosts on the tunnelled network is sent through the tunnel. All communication intended for other hosts is sent normally through the directly connected network.

Most firewalls can be set up to permit the passing of encrypted tunnel traffic into a trusted network and on to the tunnel server. The usual approach is to relay traffic

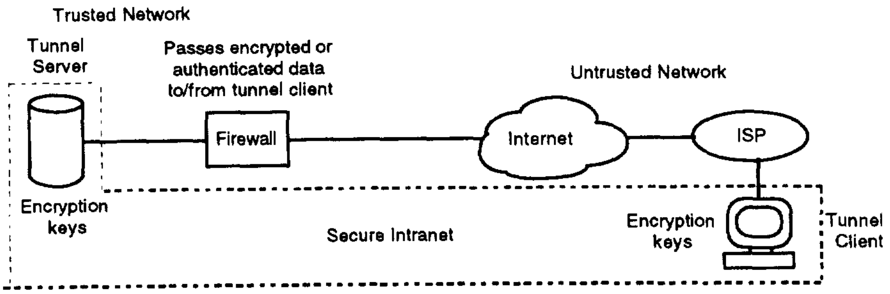

Fig. 12. Intranet architecture with encrypted tunnels.

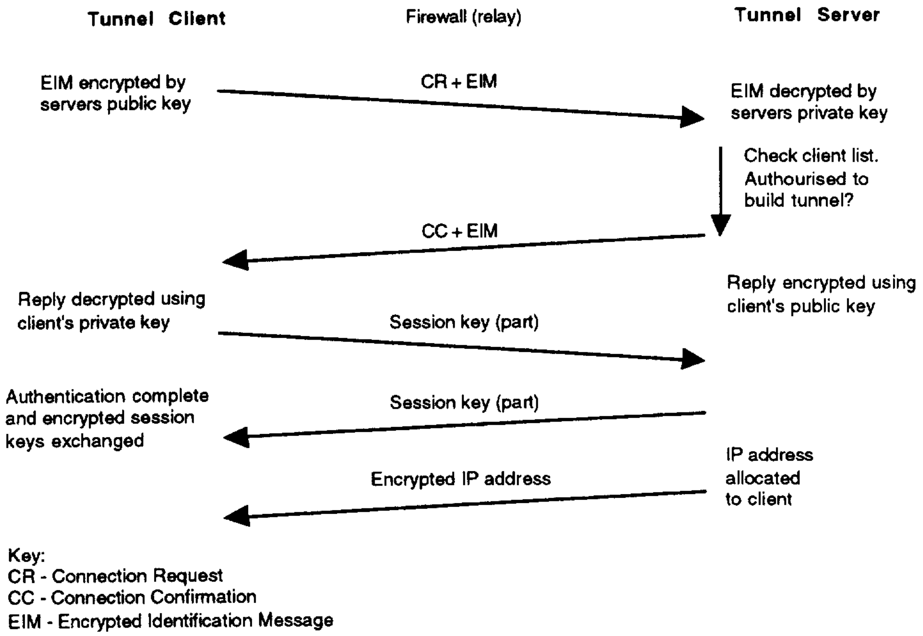
Tunnel Client
Firewall (relay)
Tunnel Server
EIM encrypted by servers public key
CR + EIM
EIM decrypted by servers private key
Check client list. Authourised to build tunnel?
CC + EIM
Reply encrypted using client's public key
Reply decrypted using client's private key
Session key (part)
Authentication complete and encrypted session keys exchanged
Session key (part)
IP address allocated to client
Encrypted IP address
Key:
CR - Connection Request
CC - Connection Confirmation
EIM - Encrypted Identification Message

Fig. 13. Authentication sequence for encrypted tunnel establishment.

addressed to a specified port on the firewall on to a specified port on the tunnel server, therefore hiding the actual address of the tunnel server as well as logging all connection attempts at both the firewall and tunnel server.

3.3. PPTP and L2F

Microsoft's PPTP [13] is an extension to the standard point-to-point protocol (PPP) that is used to create multiprotocol virtual private networks (VPNs) via local access to an ISP. Cisco's equivalent is L2F. Use of the Internet or Intranets in conjunction with secure encrypted transactions creates VPNs at a fraction of the cost of conventional network services.

The VPN can be viewed as an application, whereas PPTP or L2F are the protocols in the same way that e-mail is an application and SMTP is the protocol used. As an extension to the PPP, PPTP can only handle point-to-point and not multipoint links.

The result can be global networking with substantial economies of scale. In addition, no changes need be made to existing LAN applications. PPTP/L2F is designed to tackle the twin concerns of cost and security, using the Internet as an inter-LAN technology and opening up the benefits of low-cost and secure communications. PPTP is supported by Microsoft, 3Com, Ascend, Telematics and US Robotics. The competitor to PPTP is Cisco's PIX Private Link product for VPNs and is based upon L2F. Suggestions have been made to combine the best features of L2F (Cisco) and PPTP (Microsoft) into a new protocol called L2TP.

3.4. SSL

SSL provides a method for adding security to existing applications by incorporating RSA data security technology. SSL is application protocol independent and provides

- encryption (which creates a secured channel)
- authentication
- uses X.509 certificates [14] and digital signatures to verify the identity of parties
- message integrity (which ensures that messages cannot be altered en route).

The RSA encryption technology used by both SSL and encryption tunnelling is similar, although the encryption is done at different layers of the architectural stack. SSL encrypts data at the application layer while tunnels establish a link for all logical connections between the two networks. Applications which need to encrypt a specific session, such as a web browser, Telnet or FTP session, must be modified to enable the request for an encrypted link. With tunnelling, the applications do not have to be modified as all traffic between tunnelled networks is encrypted.

Netscape Communications has designed the SSL [15] protocol for providing data security between application protocols (such as HTTP, Telnet, NNTP or FTP) and TCP/IP. SSL is designed to provide privacy between client and server applications on the Internet. It provides data encryption (using a 40-bit key for the RC4 stream encryption algorithm), server authentication, message integrity, and optional client authentication for a TCP/IP connection. Further, SSL requires a reliable transport protocol such TCP for data transmission and reception.

One advantage of the SSL protocol is that it is application independent. Higher-level application protocols, such as HTTP, FTP, Telnet, layer transparently on top of SSL which can negotiate an encryption algorithm and session key as well as authenticate a server before the application protocol transmits or receives data. All transmitted application protocol data is encrypted, thus ensuring privacy.

SSL provides a security ''handshake'' that is used to initiate the TCP/IP connection. This handshake results in the client and server agreeing on the level of security they will use, and fulfils any authentication requirements for the connection. Thereafter, SSL's only role is to encrypt and decrypt the application data being used (e.g. HTTP, NNTP, or Telnet). This means that all the information in both the HTTP request and response are fully encrypted, including the requested URL, any submitted form contents (including credit card numbers), any HTTP access authorisation information (usernames and passwords), and all the data returned from the server to the client.

SSL provides three fundamental security services, all of which use public-key encryption:

Service	*Underlying technology*	*Protection against*
Message privacy	Encryption	Eavesdroppers
Message integrity	Message authentication codes (keyed hash functions)	Vandals
Mutual authentication	X.509 certificates	Impostors

SSL therefore provides channel security which has three basic properties:

- the channel is private—encryption is used for all messages after a simple handshake is used to define a secret key;
- the channel is authenticated—the server endpoint of the conversation is always authenticated, whereas the client endpoint is optionally authenticated;
- the channel is reliable—the message transport includes a message integrity check.

In addition the latest version of SSL has a number of protocol enhancements including:

- fewer and faster handshake messages;
- support for more key-exchange and encryption algorithms (in particular, Diffie–Hellman and Fortezza);
- support for hardware tokens in the form of Fortezza cards. This is the first step towards more general support for cryptography-capable smart cards;
- an improvement to the client certificate request protocol that allows a server to specify a list of certificate authorities that it trusts to issue client certificates. This frees the user from having to choose a certificate for each connection.

SSL has been submitted to the W3C* working group on security for consideration as a standard security approach for WWW browsers and servers on the Internet.

Netscape Navigator supports a new URL access method, HTTPS, for connecting to HTTP servers using SSL. HTTPS is simply SSL underneath HTTP (Fig. 11). HTTP + SSL (or HTTPS) and HTTP are different protocols and typically reside on different ports (443 and 80 respectively). The same server system can run both secure and insecure HTTP servers simultaneously. For example the ''store-front'' and merchandise catalogue could be insecure, while the ordering and payment documents could be secure.

Another variation on the same theme is S/HTTP—a security-enhanced variant of HTTP and also proposed to the W3C security working group. SSL/HTTP and S/HTTP have different motivations. Whereas SSL layers security beneath application protocols like HTTP, Telnet and FTP, S/HTTP adds message-based security to HTTP specifically, drawing on the approaches of privacy enhanced mail (PEM) and multipurpose Internet mail extensions (MIME). SSL and S/HTTP are not mutually exclusive—rather they can coexist in a complementary fashion by layering S/HTTP on top of SSL.

3.5. Fortezza identification and authentication

Fortezza [16] is a PCMCIA type II card and is a key component of the National Security Agency's ongoing program for secure Department of Defence messaging. It is an integrated security solution that meets the needs of civilian and military Government agencies who require the protection of sensitive but unclassified information.

Fortezza combines a user authentication technique, which defies attempts at password sniffing, with the strength of the Digital Signature Algorithm, proposed by the National Institute of Standards and Technology (NIST). It ensures that remote identification is not susceptible to eavesdropping. Fortezza follows the guidelines of Federal Information Processing Standard (FIPS) 185. Encryption and key management are automatic, and never require user intervention.

3.6. IPv6 security, IP security (IPSec)

The Internet has a number of security problems and lacks effective privacy and authentication mechanisms below the application layer. IPv6 or IPSec [17] remedies these shortcomings by providing two security options that may be used singly or together to provide differing levels of security to different users.

The first mechanism, called the IPv6 Authentication Header, is an extension header that provides authentication and integrity (without confidentiality) to IPv6 packets. Although the extension is algorithm independent and will support many different authentication techniques, the use of MD5 is proposed to help ensure interoperability within the worldwide Internet. This can be used to eliminate a

*The World Wide Web Consortium (W3C) is an international collaboration between research organisations, whose goal is to guide the evolution of the World Wide Web. The Consortium works with member companies to refine and revise the technical specifications of the Web as well as to create pre-competitive joint projects that extend the application of the Web into new domains.

significant class of network attacks and, in particular, host masquerading. Its placement at the IP layer can help provide host origin authentication to those upper layer protocols and services that currently lack good protection. This mechanism provides authentication and integrity, but specifically does not provide confidentiality.

The second security extension header provided with IPv6 is the IPv6 Encapsulating Security Header. This mechanism provides integrity and confidentiality to IPv6 packets. It is simpler and more flexible than some similar security protocols and is algorithm independent. To achieve interoperability within the global Internet, the use of DES is being used as the standard algorithm for use with the IPv6 Encapsulating Security Header.

3.7. Secure wide area networking (S/WAN)

S/WAN describes the concept of secure IP wide area networking, the standards for which are based upon IPv6 and IPSec developed by the IETF (Internet Engineering Task Force). S/WAN is also designed to ensure interoperability among firewall and TCP/IP-based products. S/WAN also uses the IETF's security architecture for the Internet protocol, RFC 1825–1829 [18]. It supports encryption at the IP level using RSA (asymmetric) or DES (symmetric) encryption with keys ranging from 40 to 128 bits. This provides more fundamental, lower-level security than higher-level protocols, such as SSL and S/HTTP. However, SSL and S/HTTP can be layered on top of S/WAN implementations so that these security specifications can work together.

S/WAN can be implemented using RSA's BSAFE [19] cryptography engine which provides software developers with a selection of public-key, secure-key and cryptographic hashing algorithms for authentication applications. BSAFE includes modules for popular encryption techniques, such as RSA, Triple DES, RC2, RC4, and RC5, and also supports digital signatures and certificates.

S/WAN's goal is to use a selection of these security features in conjunction with TCP/IP to build Internet-based VPNs. VPNs provide privacy and authentication on public networks by encrypting all data packets between communicants. VPNs can be formed between two or more firewalls across the Internet and, since it is common for organisations to have firewalls at multiple sites, it is a natural evolution for these firewalls to become points for encrypting communications. In the past, users have often been locked into a single-vendor solution, because vendors have been unable to agree upon the implementation details. This S/WAN effort should remove a major obstacle to the widespread deployment of secure VPNs.

Key vendors supporting the S/WAN initiative include RSA, Bay Networks, Cisco, CheckPoint Software Technologies, Digital Pathways, IBM, Raptor Systems, Secure Computing Corporation, Sun Microsystems, Trusted Information Systems, and VeriSign.

3.8. Secure electronic transactions (SETs)

In response to the demand for electronic commerce, MasterCard, Visa, Netscape, and a small group of key technology partners jointly developed the SET protocol as a method to secure credit card transactions over untrusted networks [20]. SET defines both the electronic payment protocol and the certificate management process.

SET will support electronic payment systems that

- provide for confidential transmissions
- authenticate the parties involved
- ensure the integrity of payment instructions
- authenticate the identity of the cardholder and merchant to each other.

The SET specification addresses the following requirements.

- Confidentiality of information—ensures that cardholder information is accessible only by the intended recipient. The SET specification uses message encryption to ensure confidentiality.
- Integrity of data—guarantees that message content is not altered during transmission. SET ensures data integrity with the use of digital signatures.
- Cardholder account authentication—verifies that the cardholder is the legitimate user of the account. SET ensures account authentication by using digital signatures and cardholder certificates.
- Merchant authentication—confirms to the cardholder that a merchant can accept bank card payments. SET uses digital signatures and merchant certificates to provide merchant authentication.
- Interoperability—ensures that any cardholder with compliant software can communicate with any merchant running compliant software. SET provides specific protocols and message formats to ensure interoperability on a variety of hardware and software platforms.

Fig. 14 shows payment system participants and their interactions that are supported by SET.

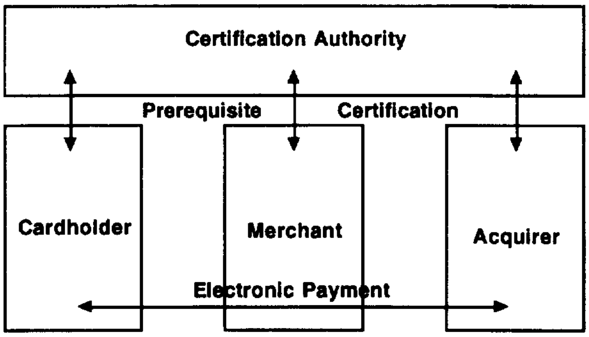

Fig. 14. Payment system participants and their interactions supported by SET.

3.9. S/MIME

S/MIME is a specification for secure electronic mail and was designed to add security to e-mail messages in MIME format [21]. The security services offered are authentication (using digital signatures) and privacy (using RSA public key encryption). More importantly, it was designed to be interoperable, so that any two packages that implement S/MIME can communicate securely.

S/MIME uses a hybrid cryptographic approach to providing security—often referred to as a digital envelope. The bulk message encryption is done with a symmetric cipher, and a public-key algorithm is used for both key exchange and digital signatures. S/MIME also uses X.509-based digital certificates and recommends three symmetric encryption algorithms: DES, Triple-DES, and RC2. The adjustable keysize of the RC2 algorithm makes it especially useful for applications intended for export outside the USA.

S/MIME, PGP (Pretty Good Privacy) and PEM all specify methods for securing electronic mail. All offer privacy and authentication services. Since PGP and PEM are quite different, they need to be compared with S/MIME individually.

PGP can be thought of as both a specification and an application. It relies on users to exchange keys and establish trust in each other. This informal method works well for small workgroups, but can become unmanageable for large numbers of users. S/MIME, on the other hand, utilises hierarchies in which the roles of the user and the certifier are formalised. This means that S/MIME is both more secure and more scalable than PGP implementations. S/MIME can also be well integrated into many e-mail applications, making it simple for users.

PEM was an early standard for securing e-mail that specified a message format and a hierarchy structure. The PEM message format is based on 7-bit text messages, whereas S/MIME is designed to work with MIME binary attachments as well as text. The guidelines for hierarchies are also more flexible in S/MIME. This should allow for both easy set-ups for small workgroups that do not need to be part of an all-encompassing hierarchy, and an easy path to move workgroups to the hierarchy that best suits their needs.

4. Conclusions

This paper has examined a range of issues that need to be addressed in firewall policy, technological choices for hardware and software platforms as well as the protocols currently being designed to secure transactions on untrusted networks.

Any corporate security policy which deals with Internet access and services, has to be flexible because the Internet itself is in a state of flux. Furthermore, an organisation's needs may change as the Internet offers new services and methods for doing business, particularly with the evolution of secure transaction services.

New protocols and services are emerging on the Internet. Using them may be of benefit to the organisation, but they may result in new security concerns. Thus policy needs flexibility to reflect and incorporate new concerns, particularly given the rapid nature of many changes in business today. New partnerships and alliances may bring new network connections and new risks. Few organisations are likely to remain entirely static.

The developments with secure transaction services have been remarkably successful in a short time. In part, this success has resulted from the open nature of protocol development, where companies which would otherwise be in competition work together to develop and fine-tune industry standards.

There is no doubt that the shape of firewall policy, the hardware and software platforms upon which these policies are implemented, as well as the protocols that drive secure transaction services on the Internet/Intranet, will continue to evolve rapidly over the next few years.

References

[1] C. Hare, K. Siyan, Internet Firewalls and Network Security, 2nd edition, New Riders Publishing, 1996, p. 128.

[2] W.R. Cheswick, S.M. Bellovin, Firewalls and Internet Security, 1994, pp. 54–74.

[3] National Computer Security Association (NCSA) Firewall Policy V 1.01, 1996. Internet: www.ncsa.com/store.

[4] Public-Key Cryptography Standards, RSA Data Security, Inc., Version 1.5, 1993.

[5] G.B. White, Computer System and Network Security, CRC Press, 1996, pp. 9–21.

[6] F. Avolio, M. Ranum, A network perimeter with secure external access, Proc. Internet Society Symposium on Network and Distributed System Security, San Diego, 1994 (also available from ftp.tis.com as / pub/firewalls/isoc94.ps.z).

[7] Computer Security Institute, 1996 Firewall Product Matrix, Computer Security Journal XII (1) (1996) 40–45.

[8] W.R. Cheswick, S.M. Bellovin, Firewalls and Internet Security, 1994, pp. 51–52.

[9] D.B. Chapman, E.D. Zwicky, Building Internet Firewalls, O'Reilly and Associates, 1995, pp. 153–154.

[10] C. Hare, K. Siyan, Internet Firewalls and Network Security, 2nd edition, New Riders Publishing, 1996, pp. 334–335.

[11] Internet: anonymous FTP from ftp.nec.com:/pub/security/socks.cstc.

[12] Virtual Private Networking over the Internet, Digital Equipment Corporation, White Paper, October 1995.

[13] Internet: Anonymous FTP from ftp://ftp.microsoft.com/developr/drg/ pptp, 1996.

[14] ITU Rec. X.509 (1993) | ISO/IEC 9594-8: 1995, including Draft Amendment 1: Certificate Extensions (Version 3 certificate). Also available from: http://info.itu.ch/itudoc/itu-t/rec/x/x500up/ s_x509_30222.html.

[15] Internet: http://home.netscape.com/newsref/ref/128bit.html, 1996.

[16] Internet: www.sctc.com/lockout/HTML/fortezza.html, 1996.

[17] S. Deering, R., Hinden, Internet Protocol, Version 6 (IPv6) Specification, IETF RFC 1883, 1995.

[18] R. Atkinson, P. Metzger, Security Architecture for the Internet Protocol, RFCs 1825-9, 1995.

[19] Internet: www.rsa.com/rsalabs/newfaq/q173.html, 1997.

[20] Internet: Secure Electronic Transaction (SET) Specification, Version 1, 1997, www.mastercard.com/set/setdraft2, www.visa.com/cgi-bin/vee/nt/ecomm/set/main.html and www.setco.org.

[21] Internet: www.rsa.com/rsalabs/newfaq/q131.html, 1997.