# *Network Security and SunScreen SPF-100*

*Technical White Paper*

Please
Recycle

# *Contents*

# *Introduction* 1

## *Introduction to Network Security*

The purpose of early computer networks was to share data and facilitate research among universities, government research and development institutions, and military installations. As early network protocols developed, little attention was paid to security issues because sites allowed onto the network were assumed to be trustworthy and existed to share research and data. However, as commercial organizations joined the interconnected public networks to take advantage of its resources and services, they didn't necessarily want to allow external access to all of their own resources and information. But, people intent on stealing resources or causing harm had little to prevent them from doing so. The field of computer network security emerged to meet the evolving needs of securing private networks and their assets while maintaining the benefits of connection to a worldwide network.

A basic security system, a *network security system* (now commonly known as a *firewall* system), was developed to sit between the public network system and an organization's internal network(s), or between different sections of an internal network. a network security system is really a collection of components that *control* traffic flow between networks. Such network security systems permit or deny network traffic according to an organization-defined security policy. There have been many network security implementations varying in complexity and efficiency. This paper discusses the most common components and approaches.

# 1

As the field of network security continues to grow, it reaches beyond simply protecting internal resources. Network security today must also provide additional services such as user authentication, privacy of communication, and even mechanisms for securely managing traditional business transactions (such as placing orders and billing).

## *The Need for Security Measures*

Many people, often referred to as *hackers* (or crackers), are adept at breaking into other networks and hosts. They do this to steal or corrupt data and resources, to impersonate an internal host and gain access to other networks, or to vandalize. Sometimes well-intentioned users might accidentally expose corporate data or services from within a network, especially if there is no corporate security policy. A good network security system can provide protection against malicious outsiders and prevent the accidental release of proprietary data and services without requiring every internal user to become a security expert. When such a system is in place an organization can enjoy the benefits of public network access while feeling more secure that their internal resources are protected.

## *Security Policy Design Considerations*

The first task in designing a network security system is to assess the needs and goals of the organization and create a security policy. The security policy reflects the organization's reasons for connecting to a public network, and will dictate which services it will offer internally and to the public. The first and most important decision encountered when formulating a security policy is deciding whether to *allow access to all services unless expressly denied, or to deny access to all services unless expressly permitted*. This is a crucial and sometimes overlooked step in setting up a security policy. The first case, *allow all services unless explicitly denied*, allows the most services with the least amount of intervention. The security problem is that new, dangerous services might become available and cause problems until the administrator explicitly disables them. On the other hand, when the policy is to *deny all services unless expressly allowed*, new useful services may become available but they will blocked to users. In this case the users will need to tell the administrator of the existence of the new service, the administrator will need to evaluate it, and decide whether or not to allow it.

After this fundamental decision is made, the organization then determines which services to provide to internal users and which to provide to outside networks. The design must also provide a way of monitoring security and enforcing the policies.

When designing corporate security policy and choosing a network security system, it is important to consider the cost and convenience trade-offs. Depending on the amount of security desired and the security system chosen, additional hardware may be required (including routers and dedicated hosts), special software may be needed, and a security expert may be needed to program and maintain the system. Another cost to consider is that of the effect the security system will have on productivity and service availability. Some network security system implementations will noticeably slow network speed. Others will restrict or deny useful network services such as mail and file transfers. Some require new software to be distributed to every host on the internal network, amounting to extra work for system administrators and inconvenience for users. A common goal of network security system designers is for a *transparent* security system. Such a system provides security without significantly affecting network performance, or forcing users to either give up services or learn new ones.

Another important factor in designing a network security system is balancing security and complexity. A general principle in network security design is the more complex the security system, the more vulnerable it is and the more difficult it is to maintain. a network security system increases in complexity as more networks are added and administered, as additional hardware components are added, and as the number of *screening rules* (discussed later in this paper) increases. Complex systems are difficult to configure correctly which can lead to the creation of security problems.

In summary, factors to consider when creating a security policy are:

- which services to provide internally and externally
- initial and ongoing financial investment (new hardware, software, personnel)
- convenience and service availability
- balance of complexity and level of security
- network performance

## *Beyond Basic Security: Privacy, Authentication, and Advanced Services*

Other problems are emerging in the on-line business world besides protecting the data on private networks. These dangers include competitors and hackers obtaining, diverting or altering electronic business information. This can have serious implications for those conducting business over a network. For example, the integrity of a business can be seriously compromised if the confidential messages it sends are intercepted and read by a competitor (or hacker), altered and sent to the originally intended receiver, or diverted and sent to an entirely different address. The same hazards also apply to customers who place orders using a credit card or perform other transactions over a public network. A robust network security system can provide features such as *privacy* and *authentication* to prevent these attacks.

Figure 1-1 illustrates a hacker listening to a confidential business transaction. In this example, the hacker modifies the data and sends it on to its destination, while saving data stolen from the transaction for future use.

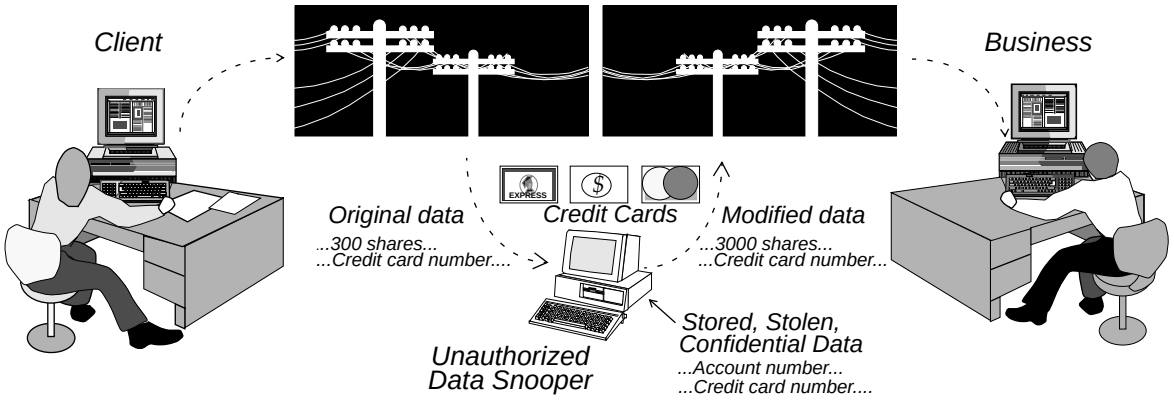


*Figure 1-1* Example of a thief stealing, storing and modifying data

*Privacy* ensures that when a message is sent, no one but the intended receiver can read or interpret it. It prevents eavesdropping, and allows confidential or proprietary information to be transferred *over a public* network securely.

*Authentication* means that the recipient of a message (mail, data, files, etc.) is assured that the sender is who the message claims, *and* that the data has not been modified during transmission.

In addition to security, privacy and authentication, a comprehensive network security system can provide mechanisms for automatic on-line billing, placing orders, accounting, and other traditional business tasks.

# 1

# Common Security Solutions 2

## Security Solutions

Several approaches can be taken when implementing an organization's security policies. They include various configurations of *packet screens*, *application gateways (or relays)*, and *De-Militarized Zones*. They are usually used in combination, with the further use of *encryption* in more advanced systems for additional levels of security.

### TCP/IP Overview

Because the type of networks that would use the network security system technology described in this paper use mostly TCP/IP protocols, a short introduction to the terminology and concepts of TCP/IP networks is provided. Figure 0-1 shows the relationship between the most common elements of TCP/IP networking.

*IP*: Internet Protocol is the standard, low-level protocol that defines the IP datagram (packet) as the unit carrying information passed through networks. This is the bottom layer in Figure 2-1.

*TCP:* Transmission Control Protocol is the standard transport-level protocol (on top of IP) on which many higher level application protocols are built. TCP is a connection-oriented protocol. In connection-oriented protocols, lost or damaged packets are retransmitted and re-ordered if necessary to match the original sequence.

*UDP:* User Datagram Protocol is a standard IP transport level protocol for sending datagrams between machines. UDP is different from TCP in that UDP is connectionless. There is no error correction, retransmission or re-ordering of packets. It is easier to falsify UDP packets than TCP packets since there are no connections or sequence numbers, and extra security is necessary when allowing UDP packets through the gateway. Most security systems disallow UDP packets for this reason.

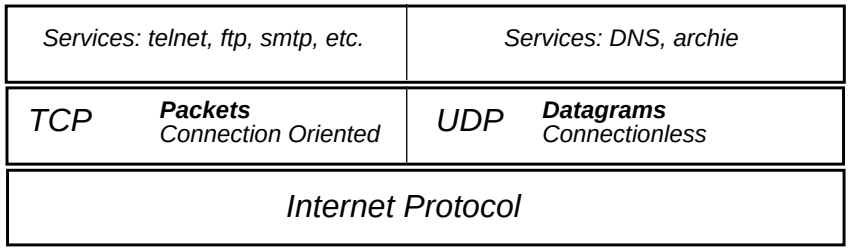


*Figure 2-1* TCP/IP and UDP

## *Packet Screening*

A packet screen is a database of rules residing in a router or special host system, (usually between two networks), that examines network traffic and decides whether or not to allow the traffic to pass, (usually from one network to the other). A packet screen will pass some data packets and block others depending on information in the packet meeting specified criteria. The criteria are specified as a set of *logical rules* (known as screening rules) and applied to each packet. Screening rules tell the screen what kinds of services to allow. For example, there might be a rule that said *allow all telnet traffic between hosts A and B from 9 am to 5 pm.* Traditionally, the information available to screen each packet on was limited to source and destination IP addresses, packet type (TCP or UDP) and destination port. The inability to look at the source port caused many packet screens to be inefficient and incomplete. Modern, more powerful packet screening engines can extract more information from the data packets and therefore can apply a more complete set of rules to incoming and outgoing packets. This increased screening ability has an increased cost of complexity, however.

Traditional packet screening implementations have certain strengths: they are fast, transparent to users and applications, are relatively protocol independent, and provide a single point (*choke point*) through which all network traffic must pass. Choke points are useful to security administrators because they provide a single, well-defined place to monitor and log traffic and enforce the security policies.

Traditional packet screening also has inherent weaknesses. They are complex to implement and maintain. Sometimes there are insufficient filtering criteria to make effective screening decisions. Often there is no context or state information about packets that previously passed through the screen.

Most routers allow screening on either incoming packets, outgoing packets, or both. Additionally, they can screen on the way into the router or the way out of the router or both, regardless of where the data is coming from. The most important reason for screening incoming packets on the way *into* the router is to protect against address spoofing (forging a fake address on a packet for subversive reasons), because crucial information such as which wire it came in on is lost at the outgoing screen. Figure 2-2 shows where packet screens may be implemented in routers.

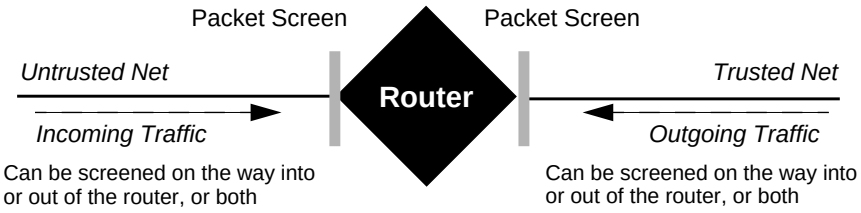


*Figure 2-2* Packet Screen Placements in Routers

Packet screens can be implemented inside the hardware router connecting two networks (trusted and untrusted), or they can be on a general purpose computer system, or "host" system, usually embedded in the operating system. The danger of using a host is that the host can be broken into and the security compromised. This danger is specific to host system packet screens because hosts usually store vital information that can be stolen, and also have facilities for gaining entrance into the system (login, telnet, etc.), and once an intruder is in, the entire network is vulnerable. However, a host system packet screen is usually more capable than a packet screen as part of a router because it is does not

have the limitations of a router. Todays routers cannot retain state, cannot perform logging functions, and programming rules in routers is usually very complicated. Also, enabling packet screening in the router slows the router down considerably.

However, since routers are the standard means of connecting networks (most sites with networks already have at least one), and they often come with basic packet screening facilities, they are the most common place to implement a packet screen.

Whether on a router or a host, when the security administrator decides on a security policy (defining what hosts and services are allowed, what hosts and services aren't allowed, and when to allow or deny traffic), the issue of complexity becomes critical. The rules describing the policy must be written in the proper syntax, with the proper logical expressions and filtering criteria, and in the proper order. *Any* mistakes in defining the rules can open security holes. The greater the number and complexity of the rules comprising a screen, the more chance the screen will act in unpredictable ways or open security holes. For example, there may be a contradiction in the rules themselves:

rule 1: allow all traffic from host A
rule 2: drop all traffic from host A

This type of contradiction can and does occur, lost in the many lines of code needed to program complex screens. The two rules may be many rules apart, and may go into the screen undetected. Another problem is with rule ordering. For example:

rule 1: allow all telnet traffic between hosts A and B from 9am-5pm
rule 2: allow all traffic from host A if it's encrypted

This is not a contradiction, but can pose a problem due to the "allow" overlap which causes ambiguity. Are the rules requiring that telnet traffic be encrypted *only* from 9am-5pm, and unencrypted at other times, or vice-versa?

The order in which the rules are evaluated can also pose problems, since most filter implementations stop processing the packet on the first match. If rule 1 is evaluated first, telnet is not allowed except between 9am and 5pm. If rule 2 is evaluated first, all encrypted traffic is allowed (including telnet), even if it arrives outside the 9-5 time range.

Traditional packet screening implementations do not provide easy screen specification mechanisms (such as graphical user interfaces). The lines of code defining the rules and rulesets must be entered by hand. Once in place,

administrators often have little assurance that the screens are correctly and completely specified. With no facilities checking for errors, ambiguities and contradictions, the administrator of such a packet screen can only wait for problems to appear before fixing them, and then it may be too late. Robust screen specification tools can do such error checking before installing the packet screen.

A significant drawback of many packet screens is the inability to retain detailed information (known as *context* or *state* information) about packets that have passed through. If information can be recorded and maintained about the packets, such as where the packets came from, where they were going, and what they were doing, more powerful and secure screening can be performed. This is especially useful for dealing with connectionless protocols. For example, when a screen sees UDP packets (a connectionless protocol), it cannot distinguish between the original request (coming from inside) and the response. The only safe way to allow connectionless protocols through the screen is to keep state information that will record the fact that a request was made, and check incoming UDP packets to see if they are expected. Packets not on the list as being expected are dropped. With state information available, a *virtual connection* can be established.

## *Application Relays*

Packet screening uses a general, protocol and service independent mechanism to screen all traffic. Application relays instead use custom, protocol or service specific software for each service provided. Every service and application, such as FTP, mail, or telnet, is usually required on the end hosts and the gateway host (the machine acting as the relay between the trusted and untrusted networks). Figure 2-3 shows an example of an application relay configuration. When the end host invokes the service, the service connects to a gateway machine (sometimes called a *Bastion Host*), which in turn connects outside. There is no forwarding of IP packets from within the internal network. All outgoing packets have the address of the application relay machine, effectively hiding the topology of the internal network.

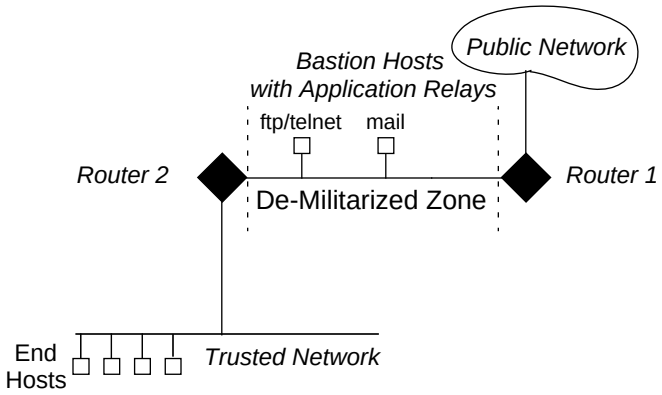


*Figure 2-3* Typical Application Relay Configuration

The advantage of application relays over packet screens are that there are no complicated rule-set interactions to worry about. Potential security holes on individual hosts won't be exposed because the individual hosts on the internal network are invisible to the external network, and incoming and outgoing traffic is easy to log and analyze on the gateway machine. Using a gateway machine to relay packets can also be used to hide internal network topology, giving away less information for use by potential hackers. For example, it is common for application relays to rename all outgoing mail messages as being from *Firstname.Lastname@Org.Domain*, instead of *username@host.domain*.

The disadvantages of application relays are that software needs to be written for each application, they are not transparent to end users (users need to learn and use new software), and the number of applications and services provided is limited by the resources available to modify and maintain them. Network speed can also suffer because of the extra connections being made and because data is being copied between the end host and the gateway, the gateway and the external host, and back again. What application relays gain in the security of not having internal hosts talk directly to external hosts, they lose in service availability and flexibility. In the long run they can also become complicated as more routers and bastion hosts are installed, configured, and maintained.

For additional security, the gateway machine itself can be placed between two routers with basic packet screening capabilities. The application pictured in Figure 0-3 is located on its own secure network between two routers. This secure network is often called a *De-Militarized Zone* (DMZ). DMZs are useful

places for locating bastion hosts containing information or services that the organization wants to allow the public access to without jeopardizing the security of the internal networks.

The pictures in Figure 2-4 through Figure 0-7 show a schematic look at the evolving complexity and functionality of network security configurations. The configurations pictured have increasing levels of security and complexity.

Figure 2-4 shows a network with minimal security. There are two networks, (public and internal) and two interfaces to administer. There is one hardware device (the router) with minimal screening facilities. There are no secure application gateways and no facilities for logging.

- **No Logging**
- **Minimal Screening Facilities in Router**

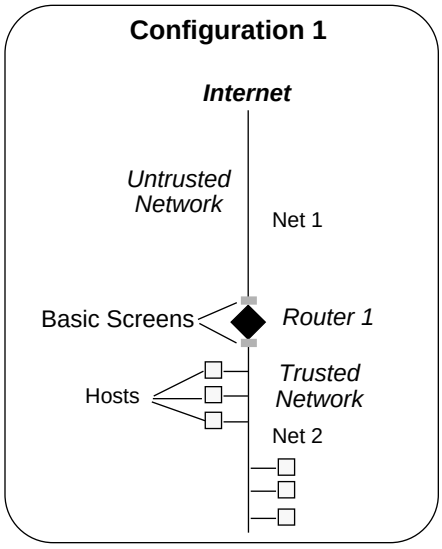


*Figure 2-4* Example of Primitive Network Security

Figure 2-5 is slightly more secure than Figure 2-4. There is a DMZ where application relays and public information can be kept. There are three networks to manage, three interfaces, and no useful logging of packets reaching the secure network. This scenario is relatively inexpensive with only one hardware device needed.

- **De-Militarized Zone**
- **Increasing Number of Components to Manage**
- **No Logging**

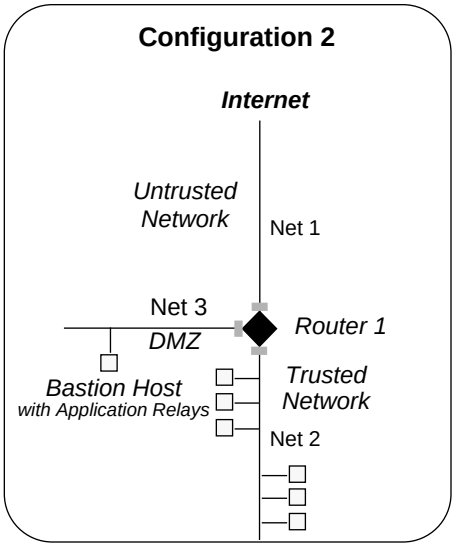


*Figure 2-5* Intermediate Network Security with DMZ

Figure 2-6 is even more secure with two routers to go through before reaching the internal network. There is a DMZ where application relays and public information can be kept. There are three networks and four interfaces to administer. Two machines are needed to do logging, one on the DMZ and one inside the internal network. Both sets of data are then compared to come up with useful information. Due to the cost of routers, this scenario is much more expensive.

- **De-Militarized Zone**
- **Two Routers, Preferably from Different Manufacturers**
- **Logging Available but Difficult, Requiring Two Separate Points**
- **Costly in Terms of Components and Maintenance**

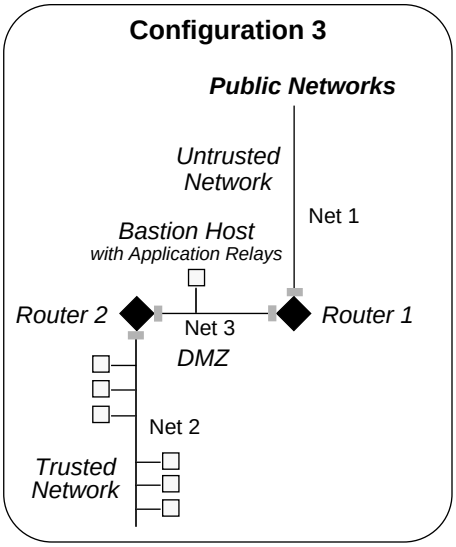


*Figure 2-6* Double Router Security with DMZ

Figure 2-7 is the most secure but also most complex of the traditional network security scenarios. It involves up to five networks and three interfaces to administer, three hardware devices (router and two chokes), and two machines do logging. For each network involved, a network address must be obtained from the Internet service provider. Application relays and public information can be kept on and monitored on the DMZ. Logging is done at two choke points. The choke hosts can provide application level security because they don't have the limitations of routers.

- **De-Militarized Zone**
- **Choke Point for Logging, but Still Need Two Points to Log Both Trusted Network and DMZ Activity**
- **Up to Five Networks to Administer**

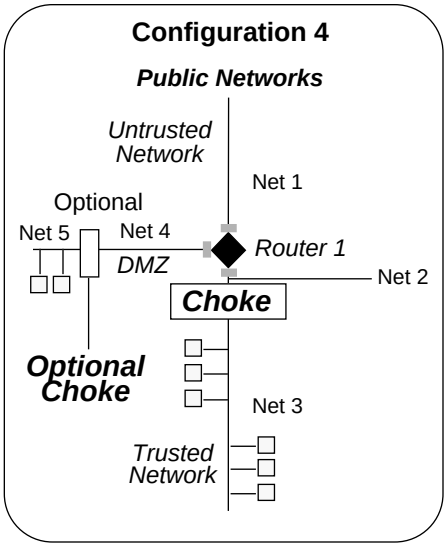


*Figure 2-7* Advanced Security with DMZ and Choke Point

In summary, the more secure the system and the more functionality offered, the more complex and expensive it becomes in terms of components to configure and administer, hardware costs, and potentially additional personnel to maintain it.

## *Privacy and Authentication*

As mentioned earlier in this paper, more security measures are needed than simply protecting internal networks. Businesses and customers need assurance that their transactions and confidential information will stay confidential and not be intercepted or altered, and that the persons with whom they are dealing are really who they say they are. This section gives a brief introduction to cryptography and how it can be used to provide privacy and authentication in online and networked business endeavors. For a more detailed discussion of cryptography, please refer to ICG's *Cryptography in Public Data Networks with SunScreen* Paper.

Cryptography uses mathematical formulas to encrypt data into a form unreadable without a decryption key. In the most basic sense it provides a level of privacy by rendering information useless to anyone without the decryption key.

In a more advanced process it can be used to provide authentication. There are two common methods of using encryption for privacy and authentication purposes: shared key cryptography and public key cryptography.

In shared key technology, the same key is shared by trusted parties. It is used to both encrypt and decrypt the message. It works as follows: a message (called plaintext) is encrypted using the shared key. This produces ciphertext, which cannot be read without being decrypted. The same key (or a copy of the same key) is used to decrypt the ciphertext back into plaintext again. The advantage of shared key cryptography is its speed. It is much faster than public key cryptography. The disadvantage of shared key cryptography is that it is easier than public key cryptography for untrusted parties to obtain copies of the key.

In public key cryptography, two keys are needed, a *public* key and a *private* key. The two keys are generated together using a one-way mathematical function. In effect, the public key is the inverse of the private key. Whenever a message is encrypted with one of the keys, it must be decrypted with its mate. They are used together in a different combination depending on whether privacy or authentication is the goal. The public key can be distributed either to trusted parties or it can be widely published in a directory. The private key is kept secret. Because no other copy is available, there is much less chance of an untrusted party gaining a copy of it.

When the public key method is used for privacy, the sender encrypts the message with the *intended receiver's* public key. The receiver decrypts the message using the receiver's private key. When public keys are used for authentication, the sender encrypts the message using the sender's *private key*, and the receiver decrypts using the sender's public key.

# *SunScreen SPF 100* 3

Sun Microsystem's Internet Commerce Group designed and developed a network security architecture called ***SunScreen$^{tm}$*** that provides a high level of security and functionality with minimal complexity. SunScreen$^{tm}$ combines highly advanced filtering, authentication, and privacy technology with a simple mechanism for administration to provide a powerful, easy to use security solution. It is network, protocol, and application independent. The SunScreen$^{tm}$ configuration consists of a central hardware device (SunScreen$^{tm}$ SPF-100) and a secure Administration Station from which the SunScreen$^{tm}$ security rules and parameters are specified. Figure 3-1 shows the SunScreen$^{tm}$ SPF-100 and the SunScreen$^{tm}$ Administration Station.

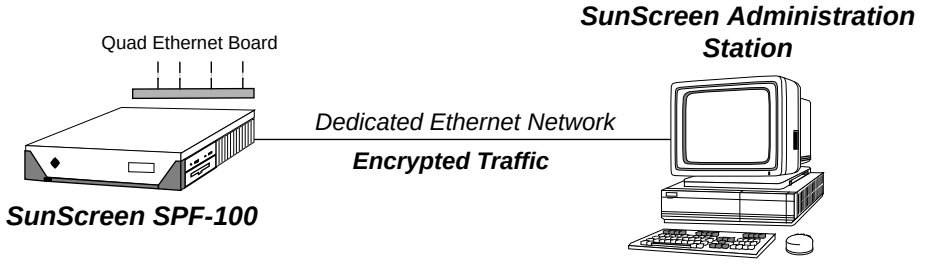


*Figure 3-1* SunScreen$^{tm}$ Architecture

The SunScreen[tm] has advanced functionality (such as stateful screening) and runs on a dedicated host system. This means it doesn't have the security problems of a packet screen running on a non-dedicated system, as will be discussed in the next section. An advanced graphical user interface (GUI) provides a simple, intuitive way to implement a security policy and define screening rules.

SunScreen[tm] is not a router, and is therefore undetectable. Packets pass through without recording any indications of its existence. This gives potential intruders less knowledge to exploit. It also allows several wires to be administered as a single network with the same range of IP addresses. This reduces the need for additional IP addresses and interfaces and provides a central place to do logging and administration.

## *Configuration*

The SunScreen[tm] is a dedicated hardware device based on SPARC technology. The operating system does not allow logins, and all standard network services, such as mail, are disabled. The packet screen configuration information (specified with a GUI) is downloaded into the device in a cryptographically secure manner. *Authentication technology* verifies both the information and the administrator. Once configured and running, SunScreen[tm] provides a central choke point for screening, logging, alerting, or even disconnecting in an emergency. It has multiple ethernet ports, with one port reserved for the administration station.

As shown in Figure 3-2, a network configuration using SunScreen[tm] has only two machines (one router and one choke, the SunScreen[tm]), and only two networks to configure and administer (although there are several wires constituting one virtual network).

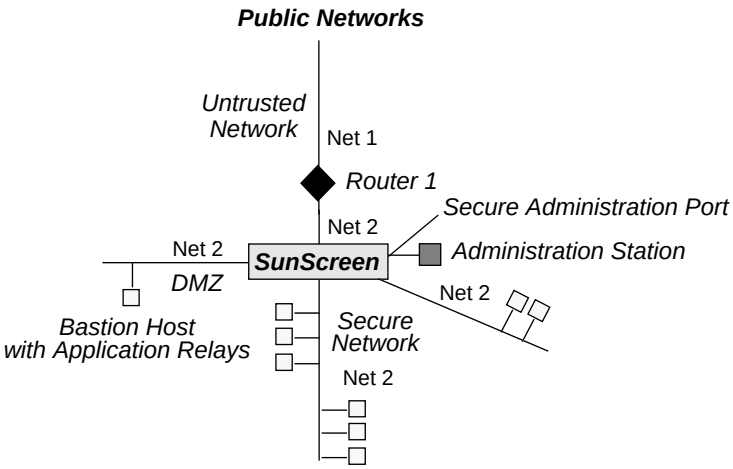


*Figure 3-2* SunScreen[tm] Potential Network Configuration

## *Packet Screening Capabilities*

The SunScreen[tm] packet screening engine screens traffic coming into and leaving the trusted network. It can extract and examine any portion of the packets, allowing for powerful rules and decision making. Actions that may be taken on packets include pass, reject, reject with a notification to the sender, encrypt, decrypt, alert, and log.

Knowledge of state is also retained, which helps allow UDP and TCP services securely. The filtering engine extracts relevant information from the packets, stores it, and uses it to make decisions on future packets. Using FTP as an example, the engine will extract the *port command* information (port number) from outgoing FTP requests, and based on this knowledge will allow an FTP data connection at the assigned port number for the duration of the connection.

## *Privacy and Authentication*

SunScreen[tm] can encrypt and decrypt packet contents at the IP packet level between other SunScreens[tm] using a variety of encryption standards. This happens transparently to the end users and applications. SunScreen[tm] uses this

encryption capability with authentication protocols and key generation and management technology to provide authentication and privacy between sites with SunScreens[tm].

For more background and details on encryption, privacy and authentication, please refer to ICG's *Cryptography in Public Internetworks Networks with SunScreen[tm]* paper (ICG-95-0002).

### *Virtual Private Networks*

Often organizations have offices that are geographically dispersed. SunScreen[tm] provides a mechanism for allowing the different offices to use public networks as a secure, private network without needing dedicated lines and with no application change. This is achieved through data *tunneling*. When using tunneling, data packets are encapsulated inside other packets and encrypted to make public networks serve as a private network. In this fashion the dispersed offices can behave and be administered as a *virtual private network*. Hosts on this network can have the same range of network IP addresses.

## *SunScreen Administration Station*

The SunScreen[tm] Administration Station is a dedicated system providing a simple, secure mechanism for specifying and administering security policies. SunScreen[tm] comes pre-configured with the most common security rules for sites that don't wish to implement a custom security policy. It is simple to configure and offers powerful tools for developing custom packet screens. A Graphical User Interface (GUI) provides a simple way to implement a packet screen and shields security administrators from the complexities and hazards of specifying rule syntax and rule sequencing. When the screens are configured, they are loaded into the SunScreen[tm] via an authenticated, private ethernet port. The authentication technology ensures that only authorized personnel can access and modify the rules.

### *Graphical User Interface*

The GUI provides an easy way to implement the security policy. The administrator does not need to learn low-level protocols or worry about rule sequencing and creating security holes. Services to be allowed are specified using the GUI, then downloaded through an encrypted channel into the filtering device.

When choosing which services to offer, it is significant to note that the packet screen works on the principle: *that which is not expressly permitted is denied.* If the administrator does not set a rule that allows specific traffic through, SunScreen[tm] will not permit it.

Using the GUI, the site administrator can modify filtering rules and encryption parameters, retrieve current configuration from the SunScreen[tm] for editing, retrieve log files, and save log files and configurations to tape. More than one SunScreen[tm] can be administered at a time from a single GUI.

# *Summary* 4

Public networks are a common and necessary part of the computing world today. They provide many important capabilities to organizations of all types and sizes, from corporate communications and collaboration, to marketing and retail commerce. Tens of thousands of companies and hundreds of millions of users now interact with and on public networks. Unfortunately, security has not kept pace with growth.

While there are many network security access control solutions available today, SunScreen[tm] SPF-100 provides broader, more robust, and more flexible network security than any other solution on the market. SunScreen[tm] SPF-100 was designed to deliver firewall protection and virtual private network support across public networks, at a fraction of the cost of competing systems. Based upon high-performance, low-cost microSPARC-II technology, SunScreen[tm] SPF-100 provides standards-based encryption, authentication, and privacy features in a turnkey solution.

SunScreen[tm] SPF-100 was also designed to provide administrators with the necessary tools to flexibly and intuitively manage their gateway access to public networks. Employing a dedicated administration station, the SunScreen[tm] SPF-100 system ensures absolute administration privacy and easy-to-use rule-based controls to ensure that internal corporate networks and inter-company communications are safeguarded.

The Internet's phenomenal growth has brought with it equal interest in electronic commerce. As companies begin examining the security issues of network-based commerce, they are discovering that only a well-defined infrastructure providing security from intrusion and protection of sensitive information will meet their needs. SunScreen™ SPF-100 provides it.

Among other factors that set the SunScreen™ apart from other network security solutions is its unique access control system providing:

- Stateful packet filtering
- Packet Tunneling (Encapsulation)
- Available Authentication and Encryption Facilities

These features, together with its network efficiency, flexible programmatic capabilities, dedicated usage, and transparency on the network make it an ideal choice for companies seeking greater use of public networks.

# *Glossary*

**Application Relay**

A device (such as a host workstation) that connects two or more networks and forwards packets between them. Operates at the application layer of the protocol stack. It is a common component of network security systems.

**Authentication**

Methods used by a receiver of data to validate that the data was generated by the expected source.

**Bastion Host**

A central machine in a network security complex serving as a safe place to offer services and log network activity. Due to the extra security measures fortifying most bastion hosts, they are good locations for application relays.

**Choke Point**

A machine (a router or packet screening host) that is the only way traffic leaves or enters a network. It is an ideal place to monitor network activity. Because it is the sole connection between two networks, it is where the majority of the network security measures can be focused.

**Datagram**

A packet containing enough information that it can be completely routed to a destination without additional information. It usually contains a complete source and destination address in addition to its data. It is sometimes called an *Internet packet*.

**De-Militarized Zone**

A network or section of a network between an untrusted network and a trusted network that has some degree of security (usually provided by packet screen) and where application relays and public information can be located.

**Decryption**

Transformation of encrypted meaningless data into meaningful data using a key.

**Encryption**

Transformation of data into unreadable, meaningless data through calculations using a key.

**Firewall**

A network security and access control system consisting of several components: *filters* (or screens), *application relays* (or gateway), and sometimes *De-Militarized Zones*.

**Fragmentation**

The process of dividing a packet into multiple smaller packets so they can be sent over a communication link that only supports the smaller size.

**Gateway**

A machine that connects two or more networks, routing packets from one to another.

**Packet**

The smallest unit of communication on a "packet-switched" network.

**Packet Screen**

A device that connects two or more networks together and forwards packets between them. Unlike a router or a bridge, a packet screen will only forward a packet if it meets specified security criteria. Packet screens can operate either at the network or physical layers of the protocol stack.

**Privacy**

Methods (such as encryption) used to protect data from being read by unauthorized parties.

**Router**

A device that connects two or more networks and forwards packets between them. A router usually refers to a device that makes forwarding decisions based on information at the *network level* of the protocol stack.

**Security Policy**

The decisions about what services will and will not be allowed to the internal users and to the public in the interests of internal security. To a lesser extent, the security policy will determine what actions will be taken against users who maliciously try to subvert the security system. The foundation of any security policy is the determination of whether “that which is not expressly permitted will be denied”, or “that which is not expressly denied will be allowed”.

**Service**

Application level programs, such as ftp, mail, telnet, etc.

**State (Context)**

In the context of network security, it means storing information about previous network traffic in order to make decisions about future network traffic.

**Tunneling**

A method where packets are encapsulated inside other packets in order to bypass security systems or routers. While the negative implications are obvious, tunneling can also be used for positive reasons, such as providing virtual private lines between two geographically distant sites.

**Vectoring**

The act of a screen passing packets to another device for further processing.

# *References*


William R. Cheswick, Steven M. Bellovin, *Firewalls and Internet Security: Repelling the Wily Hacker*, 1994, Addison-Wesley Publishing Company.

D. Brent Chapman, "Network (In)Security Through IP Packet Filtering", *Proceedings of the Third USENIX UNIX Security Symposium*; Baltimore, MD; September, 1992.

Douglas E. Comer, *Internetworking with TCP/IP*, Volume I; Second Edition, 1991, Prentice Hall, Inc.

Marcus J. Ranum, "A Network Firewall", *Proceedings of the World Conference on System Administration and Security*; Washington D.C., July, 1992.

Andrew S. Tanenbaum, *Computer Networks*, 1981, Prentice Hall, Inc.

*Cryptography in Public Data Networks with SunScreen White Paper*, Sun Microsystems Computer Company, 1995


***Look for additional papers available on the Internet**:*

research.att.com:dist/internet_security
tis.com:pub/firewalls
ftp.greatcircle.com:pub/firewalls/papers

Sun Microsystems
Computer Company

A Sun Microsystems, Inc. Business


For U.S. Sales Office locations, call: 800 821-4643
In California: 800 821-4642

Australia: (02) 844 5000
Belgium: +32 2 716 79 11
Brazil: 011-55-11-887-9011
Canada: 416 477-6745
C.I.S.: 7-502-256-5470
Finland: +358-0-525561
France: (1) 30 67 50 00
Germany: (0) 89-46 00 8-0

Greece: +30-1-689-2210
Hong Kong: 852 802 4188]
Hungary: 36-1-202-4415
Ireland: +353-1-6684377
Italy: 039 60551
Japan: (03) 5717-5000
Korea: 822-563-8700
Latin America/Caribbean: +1 415 688-9464
Mexico: 011-525-580-5229

Netherlands: 033 501234
New Zealand: (04) 499 2344
Nordic Countries: +46 (0) 8 623 90 00
PRC: 861-8492828
Poland: 48-2-658-4535
Singapore: 224 3388
Spain: (91) 5551648
Switzerland: (01) 825 71 11
Taiwan: 2-514-0567

U.A.E.: +9714-366-333
United Kingdom: 0276 20444
United States: 800-821-4643
Venezuela: 011-582-285-6640

Elsewhere in the world, call
Worldwide Headquarters:
415 960-1300



Printed in USA