IMPLEMENTATION OF INTELLIGENT FIREWALL TO CHECK INTERNET HACKERS THREAT

Roopa K.
Dept of CS and Engg.,
J.N.N College of Engineering,
Shimoga, Karnataka, INDIA -577204
email: roopasindhe@lycos.com
Ph No: 98440 02757

Panduranga Rao MV
Dept of IS and Engg.,
J.N.N College of Engineering,
Shimoga, Karnataka, INDIA -577204
email: raomvp@yahoo.com
URL: http://www.raomvp.bravepages.com/

1. Abstract:

Computer security is the process of preventing and detecting unauthorized use of your computer. Prevention measures help you to stop unauthorized users (also known as "hackers or intruders") from accessing any part of your computer system. Detection helps you to determine whether or not someone attempted to break into your system, if they were successful, and what they may have done.

This article describes the implementation of a firewall, which is a combination of hardware and software used to implement a security policy governing the network traffic between two or more networks, some of which may be under your administrative control (e.g., your organization's networks) and some of which may be out of your control (e.g., the Internet). A network firewall commonly serves as a primary line of defense against external threats to the organization's computer systems, networks, and critical information. Firewalls can also be used to partition the organization's internal networks, reducing your risk from insider attacks. It employs the Linux operating system to have the advantage of security by birth of the operating system.

The firewall's purpose is to enforce the security policies we define. These policies reflect the decisions we've made about which internet services has to be accessible to our computers, which services we want to offer the world from our computers, which services we want to offer to specific remote users , and which services and programs we want to run locally for our own private use. Security policies are all about access control and authenticated use of private or protected services, programs, and files on our computers

. The purpose of this module is to cover the fundamentals of firewall functionality (packet filtering) and the deployment process. These practices assume that the desired firewall architecture includes packet filtering as a first

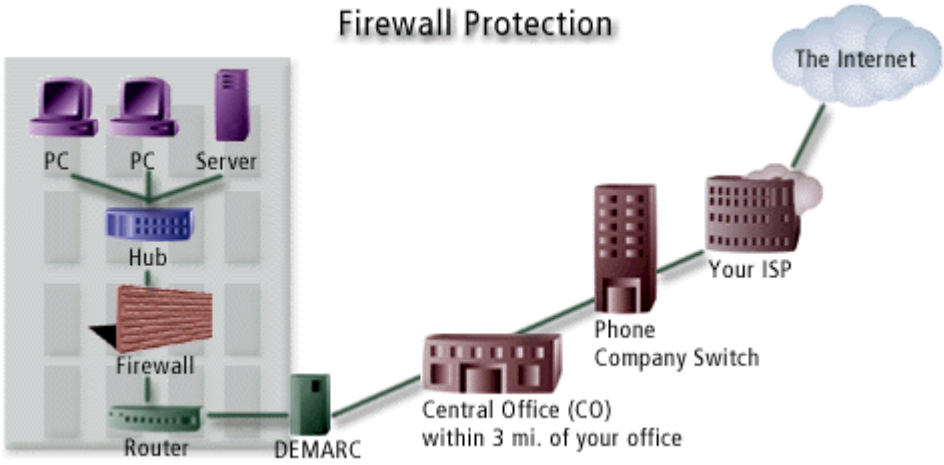
step. The article also address additional firewall capabilities such as application proxies, router, Application Level Gateways, filtering rules and service ports.

- **Key terms:** Packet filtering, Router, Application level gateway, Service ports, Application proxies.

1.Introduction

Firewall:

A component or set of components that restricts access between a protected network and the Internet, or between other sets of networks.



The use of firewall technology provides you with one of the most effective tools available to manage your networks' risk by providing you with access control mechanisms that can implement complex security policies.

To effectively deploy firewall technology, we conclude the implementation of security practices in these areas:

- preparing for firewall system deployment
- configuring your firewall system to reflect your security policy
- testing your firewall system to ensure it performs according to your specifications
- deploying the correctly configured firewall system

2.1 Packets: - IP Network Messages

The term ***pocket refers*** to an Internet Protocol (IP) network message. The IP standard defines the structure of a message sent between two computers over the network. It's the name given to a single, discrete message or price of information that is sent across an Ethernet network. Structurally, a packet contains an information header and a message body containing the data being transferred. The body of the IP pocket-its data-is all or a piece (a fragment) of a higher-level protocol message.

3. Analysis:

- **Packet Filter:** this firewall checks every packet of data and allows or denies the packet to access your network. The filter can check for content, source,

or specifications depending on your requirements. Packet filter firewalls are vulnerable to hackers who "spoof" or falsify their profile and gain access posing as someone else.

- **Stateful Inspection:** These firewalls deny "spoofed" packets by thorough inspection of all data packets. All incoming packets that arrive are checked and compared against outgoing packets for port number, source, and destination. Based on that comparison, the firewall determines if access is granted.
- **Proxy Server:** Generally considered the most secure firewall. The proxy shields your PC or company network by releasing an absolute minimum of information to the outside world. Proxy servers change the IP address heading on all outgoing data packets so they look like the originate at the firewall.

3.1 Policy considerations

The organization's networked systems security policy should include

- The risks you intend to manage with the firewall
- The services you intend to offer to untrusted networks from your protected network. These could be offerings to the Internet or to other internal networks.
- The services you intend to request from untrusted networks via your protected network. These could be requests to the Internet or to other internal networks.
- the objective that all incoming and outgoing network traffic must go through the firewall (i.e., that no traffic which bypasses the firewall is permitted, for example, by using modems) — or conversely, that specific loopholes are permitted and under what conditions (e.g., modems, tunnels, connections to ISPs)

Some typical filtering rules include:

- Permit incoming Telnet sessions only to a specific list of internal hosts.
- Permit incoming FTP sessions only to specific internal hosts.
- Permit all outbound Telnet sessions
- Permit all outbound FTP sessions
- Deny all incoming traffic from specific external networks.
- Service Independent Filtering.
- Source IP address Spoofing Attacks.
- Source Routing Attacks.
- Tiny Fragment Attacks.

4. Design and Limitations

Since routers are commonly deployed where networks with differing security requirements and policy meet, it makes sense to employ packet filtering on routers to allow only authorized network traffic to the extent possible. The use of packet filtering in those routers can be a cost-effective mechanism to add firewall capability to an existing routing infrastructure. As the name implies, packet filters specify packets to filter (discard) during the routing process. These filtering

decisions are usually based on contents of the individual packet headers (e.g., source address, destination address, protocol, port).

. In addition, adding filtering to a router

- Can negatively impact routing, and therefore networking, performance
- May require additional memory

Your ability to enforce your organization's security policies accurately can be severely impaired if you have not chosen an appropriate and effective firewall architecture. This design will determine which policies can and cannot be enforced, as well as how well the firewall will accomplish its objectives over time. Firewall architectures are difficult and expensive to change after deployment, so there is considerable value (cost savings) in creating an effective, scalable, and manageable design first.

Having chosen the basic architecture (i.e., the number of hosts, the method in which they are connected, the tasks that each will perform), the next step is to select the firewall functions to be implemented in these hosts. The two most basic categories of firewall function are packet filtering and application proxies. These functions can be used separately or jointly and can be implemented on the same or on different firewall hosts. Recently, packet filtering firewall architecture have gained some of the features of application proxies and are generally referred to as stateful inspection packet filters.

There are good reasons to use both packet filtering and application proxies. Certain services (e.g., SMTP, HTTP, or NTP) are usually safe to control via packet filters while others (e.g., DNS, FTP) may require the more complex features available only in proxies. Packet filtering is fast, while application proxies are generally slower. In cases where greater access control is required and the poorer performance of proxies cannot be tolerated, stateful inspection packet filters may be an acceptable compromise.

4.1 Limitations of Packet - filtering routers

Defining packet filters can be a complex task because network administrators need to have a detailed understanding of the various Internet services, packet header formats, and the specific values they expect to find in each field. If complex filtering requirements must be supported, the filtering rule set can become very long and complicated, making it difficult to manage and comprehend. Finally, there are few testing facilities to verify the correctness of the filtering rules after they are configured on the router.

The firewall's purpose is to enforce the security policies you define. These policies reflect the decisions you've made about which internet services you want to be accessible to your computers, which services you want to offer the world from your computers, which services you want to offer to specific remote users or sites, and which services and programs you want to run locally for your own services and programs you want to run locally for your own private use. Security policies are all about access control and authenticated use of private or protected services, programs, and files on your computers.

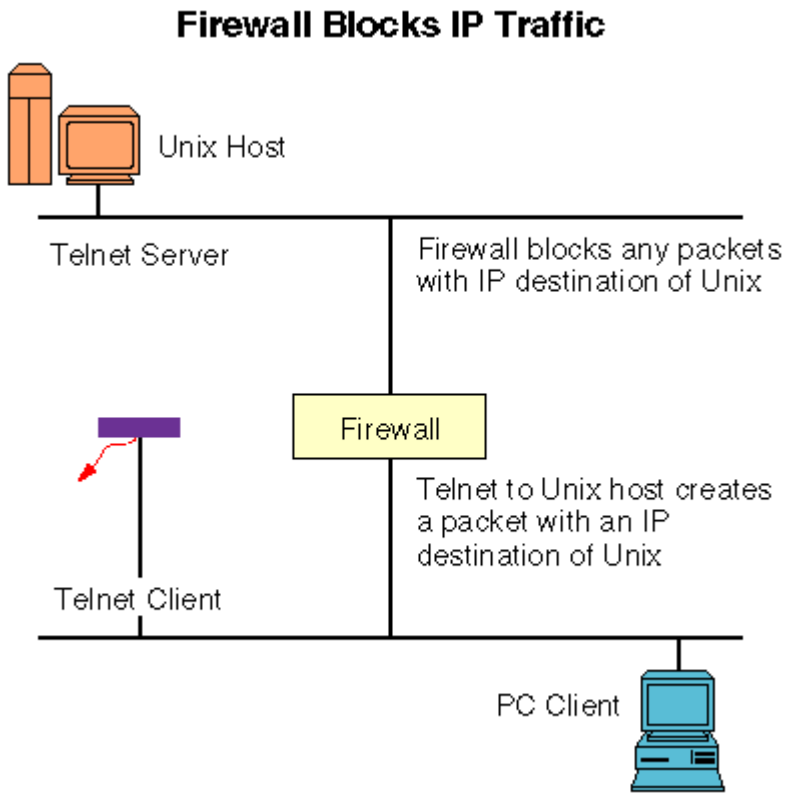
5. Implementation Details

The firewall architecture offer IP address filtering. These filters work by examining the header of the IP packet and making pass or fail decisions based on

the source and destination IP addresses. One segment has a UNIX host, and the other has a PC client

When the PC client tries to Telnet to the UNIX host, the Telnet client on the PC generates a TCP packet, and hands it to the local stack for delivery. Inturn, the stack places the TCP packet inside of an IP packet, and then sends to the UNIX host via the route defined in the PC client's TCP/IP stack. In our case, the PC client is sending the IP packet to the firewall for delivery to the UNIX host.

Suppose that we have told the firewall that it is not to accept any packets destined for the UNIX host, as depicted in below. Then the firewall would reject the IP packet, perhaps bothering to tell the client or perhaps not. Since no IP traffic for that destination would get forwarded, only users on the same segment would be able to access the UNIX host.



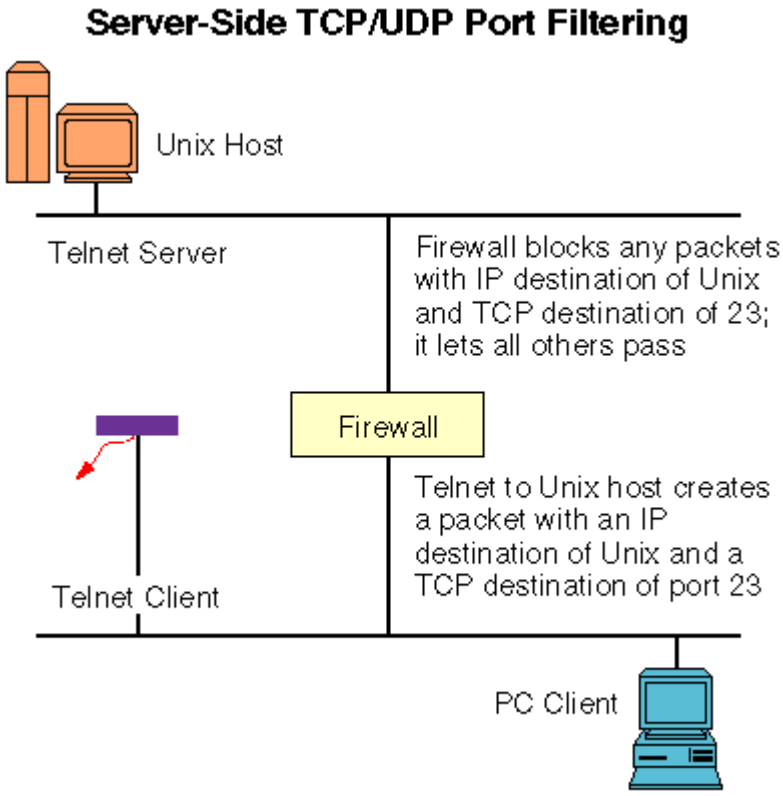
Another scenario might be that the firewall has been configured so that it simply will not accept any packets from that PC in particular. Then other systems could connect to the UNIX host, but that specific PC could not.

This type of filtering is the most basic of all. By setting accept or reject filters per IP address, these types of firewall architectures provide very basic protection mechanisms for a simple LAN. If the systems are not allowed to communicate because of source or destination IP address filters, then the packets are simply rejected.

These types of filters are commonly used in smaller shops that need to control where users can or cannot go, but beyond that they're not extremely reliable. IP addresses can be spoofed, so using these filters by themselves are not enough to stop an intruder from getting into your network. However, it is a fundamental building block of good firewall design, and is a critical component of a complete defensive infrastructure.

5.1 TCP/UDP Port Filtering

Using simple IP address comparison to allow or reject packets is a brute method of filtering. It doesn't allow for the possibility that multiple services may be running on the destination host, some of which we may want to allow users to access. For example, we may not want users to Telnet into the system, but we may want them to be able to access the SMTP/POP mail server that's running on it. To enable this level of control, we have to be able to set filters according to the TCP or UDP port numbers in conjunction with the IP address filters.



For example, the default Telnet configuration calls for the server to monitor TCP port 23 for incoming connections. Therefore, if we know that we do not want to allow any Telnet connections to our UNIX host from the PC, we can simply tell the firewall to reject any IP packets going to the UNIX host that have a TCP destination port number of 23. Since the PC's Telnet client would normally generate just such a request, the service would effectively be disabled for it.

If we wanted to reverse this example, perhaps using SMTP and POP, then we would add these services to the acceptable list for the UNIX host's destination address, and reject all other packets. Therefore, any connection request bound for the UNIX host, which had TCP destination port addresses of 110 or 25 would be allowed to pass, but no other packets would, since they wouldn't meet this "allow" condition. This would include Telnet, thereby providing the "exclude" condition.

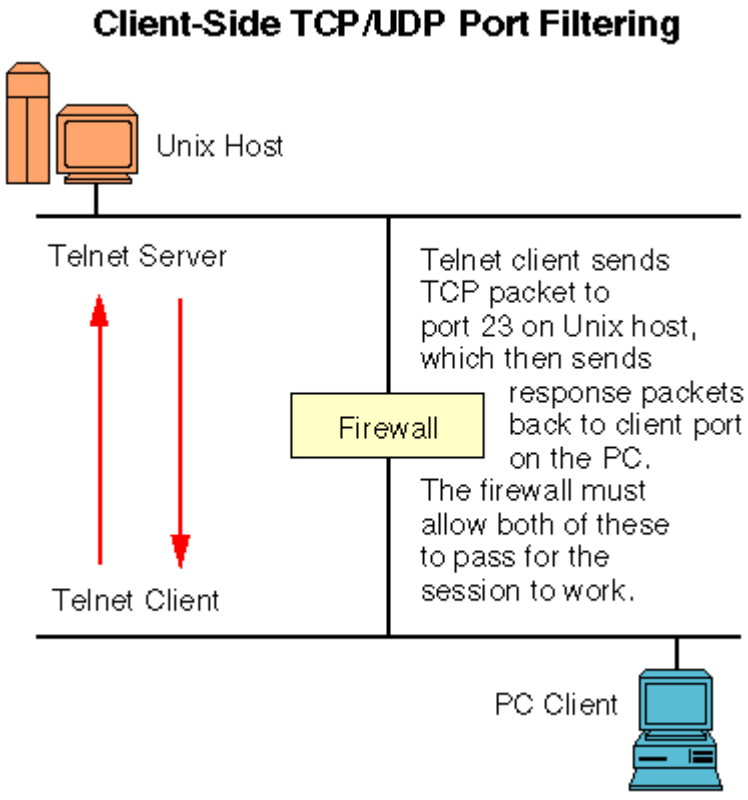
By combining the IP addresses and TCP/UDP port numbers, you can develop some pretty reliable filters. For example, if your internal SMTP mail server only talks to your Internet Service Provider's (ISP's) mail server, then you could implement a firewall filter that only allowed incoming SMTP connections that came

from the ISP's mail server, and are destined for your internal SMTP mail server. This will keep some of the hackers from being able to exploit any SENDMAIL weaknesses that you haven't plugged.

5.2 Clients have TCP/UDP Ports, Too

Since TCP/IP is a peer-to-peer protocol, each node has a unique address. This philosophy carries up into the applications layer as well, meaning that applications and services also have addresses (or port numbers). Here both the client and the server must have unique port numbers on their individual systems in order for a TCP or UDP connection to be established. For example, the Telnet server listens for incoming connections on port 23. However, the Telnet client also has a port number. Without this, the client's IP stack would not know which application the packet was for.

Historically, almost all TCP/IP client applications use a randomly assigned port number above 1023 for their end of the connection. This is a legacy from TCP/IP's roots in the UNIX world. On UNIX systems, only the root account has access to ports below 1024, which are reserved for server services (like Telnet, FTP, etc.). In order to allow client applications to work, they must use port numbers above 1023.



9. Conclusions and future scope of study:

The firewall architectures discussed enjoy levels of security in descending order: screened subnet architecture, screened host architecture, dual-homed host architecture, packet-filtering firewall and stateful inspection architecture.

Preferences also differ depending on the point of view. From the user vantagepoint, packet-filtering architecture and stateful inspection architecture are better than the application gateway because with packet-filtering firewall installed, normal activities of the internal as well as external users are less likely to be affected. The network administrator, however, would consider the application-gateway firewall architecture to be the best solution because of the ease of installation and configuration. The firewall architecture preferences of the network administrator are more likely to be in the following descending order: application-gateway architecture, packet-filtering architecture and stateful inspection architecture, screened host architecture, screened subnet architecture.

Based on the above considerations on firewall architecture, we suggest that in general:

- Small companies: Use dual-homed host or packet-filtering firewall architecture.
- Medium companies: Use dual-homed host or screened host firewall architecture.
- Large companies: Use Screened subnet architecture.

10. References and Acknowledgements:

The following references are useful to learn and implement the firewall mechanisms and to step towards cryptography.

- Building Internet Firewalls By D. Brent Chapman, Elizabeth D. Zwicky

1st Edition September 1995.

- The CERT Guide to system and Network Security Practices (addison - Wesley, 2001), by CERT author Julia Allen.
- Building Internet Firewalls, 2nd Edition

By Elizabeth D. Zwicky, Simon Cooper, D. Brent Chapman

- *Silicon Toad's Hacking Resources www.hackers.com*
- *Internet Security System www.iss.net*
- *Watchguard Firewall System www.watchguard.com*