ABSTRACT

Computers and Networking have become inseparable by now. A number of confidential transactions occur every second and today computers are used mostly for transmission rather than processing of data. So Network Security is needed to prevent hacking of data and to provide authenticated data transfer. Network Security can be achieved by Firewall. Conventional firewalls rely on the notions of restricted topology and controlled entry points to function. Restricting the network topology, difficulty in filtering of certain protocols, End-to-End encryption problems and few more problems lead to the evolution of Distributed Firewalls.

A distributed firewall is a mechanism to enforce a network domain security policy through the use of a policy language, a policy distribution scheme enabling policy control from a central point and certificates, enabling the identification of any member of the network policy domain.

Distributed firewalls secure the network by protecting critical network endpoints, exactly where hackers want to penetrate. It filters traffic from both the Internet and the internal network because the most destructive and costly hacking attacks still originate from within the organization. They provide virtually unlimited scalability. In addition, they overcome the singlepoint-of-failure problem presented by the perimeter firewall.

In our paper we deal with distributed firewall concepts, its evolution, its components, policies and a sample of the designed policy along with implementation. A distributed firewall gives complete security to the network.
1. Introduction:

Distributed firewalls are host-resident security software applications that protect the enterprise network's servers and end-user machines against unwanted intrusion. They offer the advantage of filtering traffic from both the Internet and the internal network. This enables them to prevent hacking attacks that originate from both the Internet and the internal network. This is important because the most costly and destructive attacks still originate from within the organization.

They are like personal firewalls except they offer several important advantages like central management, logging, and in some cases, access-control granularity. These features are necessary to implement corporate security policies in larger enterprises. Policies can be defined and pushed out on an enterprise-wide basis.

A feature of distributed firewalls is centralized management. The ability to populate servers and end-users machines, to configure and "push out" consistent security policies helps to maximize limited resources. The ability to gather reports and maintain updates centrally makes distributed security practical. Distributed firewalls help in two ways. Remote end-user machines can be secured. Secondly, they secure critical servers on the network preventing intrusion by malicious code and "jailing" other such code by not letting the protected server be used as a launch pad for expanded attacks.

Usually deployed behind the traditional firewall, they provide a second layer of defense. They work by enabling only essential traffic into the machine they protect, prohibiting other types of traffic to prevent unwanted intrusions. Whereas the perimeter firewall must take a generalist, common denominator approach to protecting servers on the network, distributed firewalls act as specialists.
2. Evolution of Distributed Firewall from the Conventional Firewall:

A firewall is a collection of components, interposed between two networks, that filters traffic between them according to some security policy.

**Basic structure of a firewall**

Some problems with the conventional firewalls that lead to Distributed Firewalls are as follows.

- Depends on the topology of the network.
- Do not protect networks from the internal attacks.
- Unable to handle protocols like FTP and RealAudio.
- Has single entry point and the failure of this leads to problems.
- Unable to stop "spoofed" transmissions (i.e., using false source addresses).
- Unable to log all of the network's activity and unable to dynamically open and close their networking ports.

In order to solve these problems while still retaining the advantages of the conventional firewalls, the concept of "distributed firewall" is proposed.

3. Distributed Firewall:

Distributed firewalls are host-resident security software applications that protect the enterprise network's critical endpoints against unwanted intrusion that is, its servers and end-user machines. In this concept, the security policy is defined centrally and the enforcement of the policy takes place at each endpoint (hosts, routers, etc). Usually deployed behind the traditional firewall, they provide a second layer of protection.

*Distributed Firewall*
Since all the hosts on the inside are trusted equally, if any of these machines are subverted, they can be used to launch attacks to other hosts, especially to trusted hosts for protocols like `rlogin`. Thus there is a faithful effort from the industry security organizations to move towards a system which has all the aspects of a desktop firewall but with centralized management like Distributed Firewalls.

Distributed, host-resident firewalls prevent the hacking of both the PC and its use as an entry point into the enterprise network. A compromised PC can make the whole network vulnerable to attacks. The hacker can penetrate the enterprise network uncontested and steal or corrupt corporate assets.

### 3.1. Basic Working:

Distributed firewalls are often kernel-mode applications that sit at the bottom of the OSI stack in the operating system. They filter all traffic regardless of its origin -- the Internet or the internal network. They treat both the Internet and the internal network as "unfriendly". They guard the individual machine in the same way that the perimeter firewall guards the overall network.

### 4. Policies:

One of the most often used term in case of network security and in particular distributed firewall is policy. It is essential to know about policies. A “security policy” defines the security rules of a system. Without a defined security policy, there is no way to know what access is allowed or disallowed.

A simple example for a firewall is:

- Allow all connections to the web server.
• Deny all other access.
The distribution of the policy can be different and varies with the implementation.

It can be either directly pushed to end systems, or pulled when necessary.

4.1. Pull technique:

The hosts while booting up pings to the central management server to check whether the central management server is up and active. It registers with the central management server and requests for its policies which it should implement. The central management server provides the host with its security policies.

4.2. Push technique:

The push technique is employed when the policies are updated at the central management side by the network administrator and the hosts have to be updated immediately. This push technology ensures that the hosts always have the updated policies at anytime.

The policy language defines which inbound and outbound connections on any component of the network policy domain are allowed, and can affect policy decisions on any layer of the network, being it at rejecting or passing certain packets or enforcing policies at the application layer.

5. Components of a Distributed Firewall:

1. A central management system for designing the policies.
2. A transmission system to transmit these polices.
3. Implementation of the designed policies in the client end.

5.1. Central management System:
Central Management, a component of distributed firewalls, makes it practical to secure enterprise-wide servers, desktops, laptops, and workstations. Central management provides greater control and efficiency and it decreases the maintenance costs of managing global security installations. This feature addresses the need to maximize network security resources by enabling policies to be centrally configured, deployed, monitored, and updated. From a single workstation, distributed firewalls can be scanned to understand the current operating policy and to determine if updating is required.

5.2. Policy Distribution:

The policy distribution scheme should guarantee the integrity of the policy during transfer. The distribution of the policy can be different and varies with the implementation. It can be either directly pushed to end systems, or pulled when necessary.

5.3. Host End Implementation:

The security policies transmitted from the central management server have to be implemented by the host. The host end part of the Distributed Firewall does provide any administrative control for the network administrator to control the implementation of policies. The host allows traffic based on the security rules it has implemented.

6. Policy design and implementation:

6.1. Formulation of the Policy:

User level process that makes all the decisions based on policies. Initial policies are read from a file. The implementation is done on OpenBSD. A language to express policies and resolving requests like KeyNote system is used. A “security policy” defines the security rules of a system (i.e) to decide what to allow and what not to allow. A
sample of our designed policy is mentioned as follows:

**Server side:**

1. **Server side** is passive open and listens using the `system listen()` call.
2. It accepts the incoming connections using the `accept()` call.
3. If the packets are from the undesired network (determined using the source IP address), go to decision.
4. If the incoming packets request HTTP service i.e. port no - 80 (suppose if HTTP service is to be avoided), go to decision.
5. If the packets contain malicious code, go to decision.
6. If the host (source IP address) look like an intruder, go to decision.
7. If all the conditions are overcome then permit the packets.

**Decision:**

- Deny the packets and drop them.
- Permit all other types of packets to go through.

The Policy mentioned here checks for conditions that can deny the packets and afterwards checks for permitting because if we allow permission first then all the packets may be allowed. This is similar to the usage of Access Control List (ACLs) in routers.

**Client side:**

1. **Client side** is active open and the policies are distributed here.

6.2. **State Diagram:**

6.3. **Program modules:**
Server side:

This module is the server daemon that runs at the Central management server. The server listens on a particular port for a request from the client. After accepting the connection the server daemon pushes out the security policies specific to contacting client.

Client side:

This module is executed by the client at startup. The client contacts the Central Management Server. It registers with it as an active host. It then obtains its updated policies and implements them. After implementing these security policies the traffic is monitored and controlled based on the security policies. Thus the concept of distributed firewalls is implemented.

6.4.Classes used in the implementation:

Server side:

- server_Int (Interface): Has all the methods to be implemented on the server.
- service_provider: This class implements the interface server_int.
- server: Creates an object of the service_provider class and embeds it in the registry.

Client side:

- private String calculateMacAddr(): Gets the MAC address of the machine and later sends it to the Server when accessing the object in the registry.

- public void execRules(String rules): Executes the rules distributed by the server.

6.5.Sample output:
The system is implemented in Linux operating system and the language used is Java. The Remote Method Invocation (RMI) architecture of Java is used for defining the policies in the server side and implementing the policies in the client side.

7. Conclusion:

Distributed Firewall gives complete protection to the network. It protects all the clients of the networks from the internal and external attacks. The distributed firewall system developed by us can allow or deny the traffic meant for a particular system based on the policy it has to follow. Remote end-user machines can be secured so they can't be used as entry points into the enterprise network. They secure critical servers on the network preventing intrusion by malicious code and "jailing" other such code by not letting the protected server be used as a launch pad for expanded attacks. Because the firewall is distributed across an entire network or server farm it offers unlimited scalability. The processing load is further distributed as the network grows, so performance remains high.