

# VPN Termination - Interoperation with firewalls

Institute of Communication Networks and Computer Engineering, University of Stuttgart

Student: Joao Machado > [jpmmachado@gmail.com](mailto:jpmmachado@gmail.com) <

Mentor: Dr.-Ing. Stephan Rupp

## Abstract

*Security is becoming fundamentally important in nowadays Networks.*

*There are no Universal solutions at the moment of implementing a secure Network, every different case requires a specific solution, not ideal one but optimal in the scope of the present possibilities.*

*Several technologies are deployed to assure integrity, authenticity and safety of the information traveling over the network, among those Firewalls and VPNs.*

*Firewall is a constant trade between security and functionality, as a Vpn is a secure functionality that reduces the overall security of the system. The way these mechanisms interact with each other is the objective of this paper.*

*Keywords: VPN, Firewall, SSL/TLS, IPsec, DMZ, Firewall Pinhole, Hole punching, UPnP, Socks*

## 1. Introduction

Firewalls evolved thru time in it's conceptuality, as well in complexity. Not long time ago, firewalls were operating in a black list mode, where the threats were identified and some specific ports, from the 65535 pool, were blocked. Now any firewall operates in a white list mode, where only few ports are open and heavily controlled.

Techniques to monitor open ports go from state full middleware, to proxying connections, analyzing not only the header of the packet, but also its content.

VPN's constitute a flexible and powerful mean to securely interconnect networks, or to provide secure remote access over insecure lines, as the internet.

Several issues arise on the moment of implementing a VPN solution, and yet keeping security, technical challenges pose, like crossing NAT middleware.

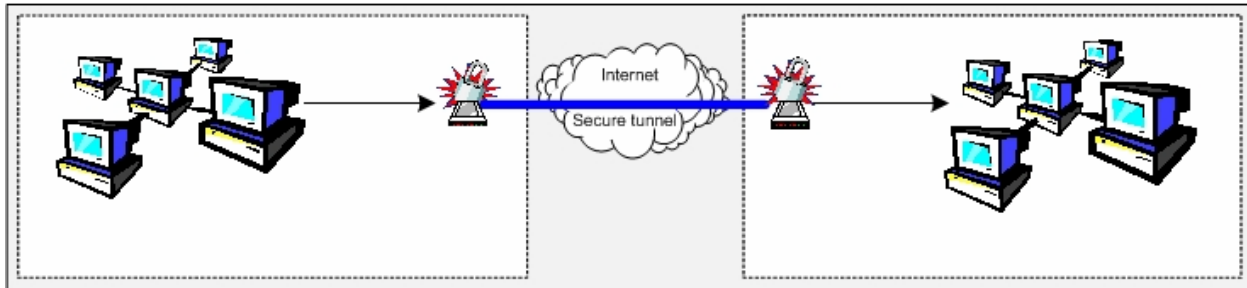
## 2.1 VPN

VPN's can be roughly divided in two types:

1. Site-to-site VPN
2. Remote access VPN

### 2.1.1 Site-to-site VPN

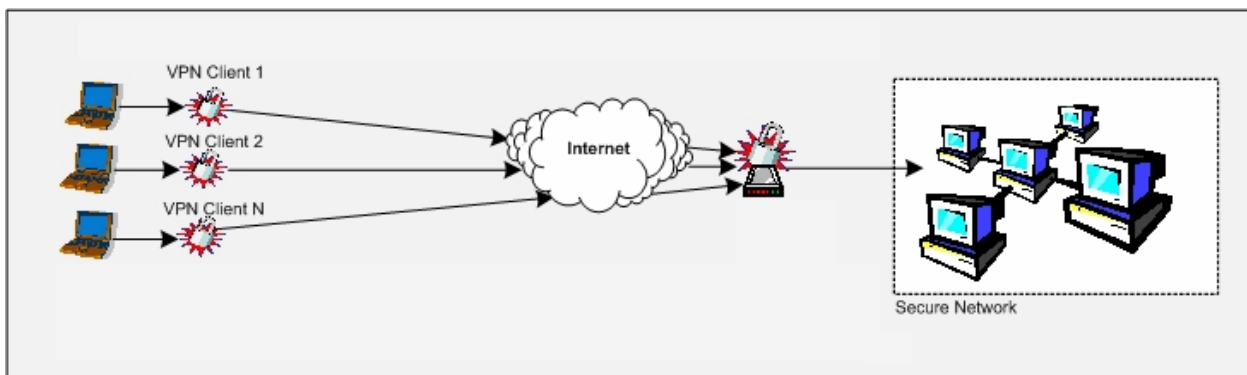
Two separate network entities are connected together over an insecure channel, where communications between both are secured (encrypted) transparently without the parties engaged in the communication being aware of it.



*Fig.1: Site to Site VPN*

### 2.1.2 Remote Access VPN

Individual end users accessing a private network over insecure public networks forming connections from their terminal to the private network.



*Fig.2: Remote Access VPN*

### 2.1.3 Requirements for VPN's

The administrator must know the extent of the VPN, being able to know at all times what data will and will not be in the VPN. Regardless of the type of VPN in use, VPNs extend the "mother network" beyond its regular capabilities, thus representing a serious security threat.

Security needs, in industry terms, authentication, confidentiality, integrity. For what all traffic on the secure VPN must be encrypted and authenticated. Many of the protocols that are used to create secure VPNs allow the creation of VPNs that have authentication but no encryption.

The security properties of the VPN must be agreed to by all parties in the VPN. Secure VPNs have one or more tunnels, and each tunnel has two endpoints. The administrators of the two endpoints of each tunnel must be able to agree on the security properties of the tunnel.

No one outside the VPN can affect the security properties of the VPN. It must be impossible for an attacker to change the security properties of any part of a VPN, such as to weaken the encryption or to affect which encryption keys are used.

Protocols must be able to cross firewalls, without compromising its security, or the security of the Network itself.

The two most widely used VPN protocols are **IPsec** and **SSL/TLS tunneling**, they work in different layers of the protocol stack, and represent different security challenges, as well Network configurations. [1]

## 2.2. IPsec

IPsec protocol suite adds a set of IP extensions that provide security services at the network level in a fashion that is compatible with the existing IP standard (IPv.4), and which is mandatory in the upcoming one (IPv.6). IPsec combines the aforementioned security technologies into a complete system that provides confidentiality, integrity, and authenticity of IP datagrams.

- IP Security Protocol proper, which defines the information to add to an IP packet to enable confidentiality, integrity, and authenticity controls as well as defining how to encrypt the packet data.
- Internet Key Exchange (IKE), which negotiates the security association between two entities and exchanges key material. IKE usage is not necessary, but it is difficult and labor-intensive to manually configure security associations. IKE should be used in most real-world applications to enable large-scale secure communications.

*RFC 2401-2411 and 2451*

IPsec creates a boundary between unprotected and protected interfaces, for a host or a network. Traffic traversing the boundary is subject to the access controls specified by the IPsec configuration. These controls indicate whether packets cross the boundary unimpeded, are afforded security services via AH or ESP, or are discarded.

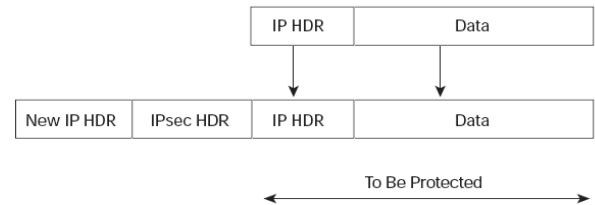
IPsec security services are offered at the IP layer through selection of appropriate security protocols, cryptographic algorithms, and cryptographic keys. IPsec can be used to protect one or more "paths" (a) between a pair of hosts, (b) between a pair of security gateways, or (c) between a security gateway and a host. A compliant host implementation **MUST** support (a) and (c) and a compliant security gateway must support all three of these forms of connectivity, since under certain circumstances a security gateway acts as a host.

The IPsec protocol suite provides three overall pieces:

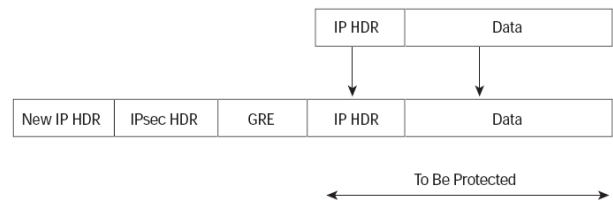
- Authentication header (AH) for IP that lets communicating parties verify that data was not modified in transit and that it genuinely came from its apparent source.
- Encapsulating security payload (ESP) format for IP that encrypts data to secure it against eavesdropping during transit protocol negotiation and key exchange protocol.
- Internet key exchange (IKE), that allows communicating parties to negotiate methods of secure communication [2].

IPsec has two methods of forwarding data across a network: transport mode and tunnel mode. Each differs in their application as well as in the amount of overhead added to the passenger packet. These protocols are summarized briefly in the next two sections:

- Tunnel Mode encapsulates and protects an entire IP packet. Because tunnel mode encapsulates or hides the IP header of the packet, a new IP header must be added in order for the packet to be successfully forwarded.



- Transport Mode IPsec transport mode inserts an IPsec header between the IP header and the GRE Header. In this case, transport mode saves an additional IP header, which results in less packet expansion [3].



### **Problem with IPsec deployment:**

IPsec increases size of large packets above supported MTU, requires fragmentation Workaround: keep remote IPsec systems out of NAT and port translating environments

- IPsec VPN clients tend to break behind firewalls even if firewall allows IPsec protocols Packet integrity checks fail if headers change between VPN gateways
- IKE problematic in NAT environments  
Workarounds: run NAT and IPsec on same gateway
- Vendor-specific: encapsulate IPsec packets over TCP or UDP
- IKE requires known source port[4]

## **2.3. SSL/TLS**

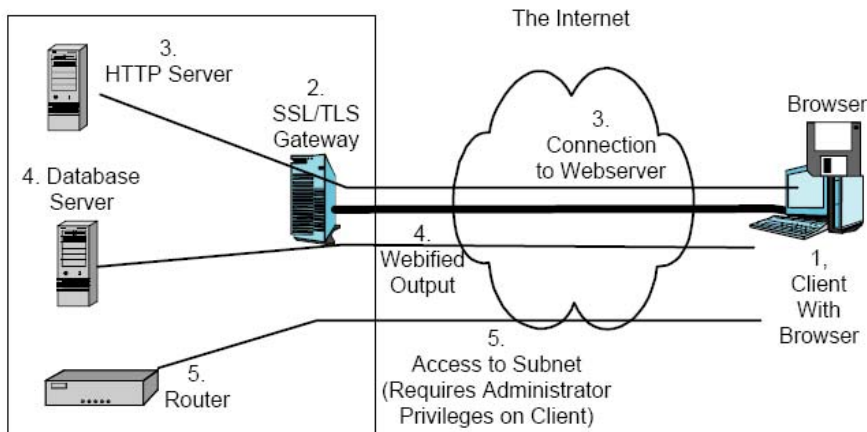
SSL runs on layers beneath application protocols such as HTTP, FTP, SMTP and NNTP and above the TCP or UDP transport protocol, which form part of the TCP/IP protocol suite. While it can add security to any protocol that uses reliable connections (such as TCP), it is most commonly used with HTTP to form HTTPS. HTTPS is used to secure World Wide Web pages for applications such as electronic commerce. It uses public key certificates to verify the identity of endpoints.

SSL/TLS is not transparent to the applications it protects. This traditionally restricted it primarily to protecting HTTP on web servers. Of course, some VPNs only need HTTP web service. In these cases, a relatively simple SSL/TLS VPN can be a good alternative to implementing a more complex VPN protocol.

Some SSL VPN's solutions don't require a VPN client software on the VPN clients. In this case SSL VPN uses the Web browser as the client application, known as "clientless" solutions. This also means the protocols that can be handled by an SSL VPN are more limited. However, this can be a security advantage. With SSL VPN's, instead of giving VPN clients access to the whole network or subnet as with IPsec it's possible to restrict them to specific applications. Some applications can't run on a browser-based, however, custom programming might create Java or Active-X plug-ins to make the application accessible through the browser. A disadvantage of this is that in order to use such plug-ins, the client's browser settings will have to be opened up to allow active content – thus exposing the browser to malicious applets unless they are set to block unsigned active content and ensure that the plug-ins are digitally signed.

SSL VPN's operate at the session layer. This gives them the ability to control access more granularly. SSL VPN's use digital certificates for server authentication. Other methods can be used for client authentication, but certificates are preferred as the most secure.

Even though there is no client software installed (other than the Web browser), SSL VPN gateways can still provide the advantages of "managed clients" by forcing the browser to run applets, for example, to verify that anti-virus software is in place before the VPN connection can be established [5].



**Fig.3 Example of SSL/TLS VPN implementation**

- The SSL/TLS VPN gateway often authenticates the client. If the client does not have a digital certificate, the VPN gateway may send the client a request for other authenticating information - usually a username and password. When the client responds, the SSL/TLS VPN gateway checks if the authenticating information is correct. Unfortunately, SSL/TLS VPN gateway authentication often is vendor-specific. So are most other aspects of SSL/TLS gateway operation.
- If the authorization succeeds, the SSL/TLS VPN gateway allows the user to connect to authorized resources within the site.[6]

In many cases, the SSL/TLS VPN gateway simply connects the client PC to a web server. This is the traditional use of SSL/TLS in VPNs. However, the SSL/TLS gateway decrypts client traffic coming into the network. This allows a firewall to check the traffic right after the VPN/SSL firewall.

In other cases, the VPN gateway connects the client PC to a database server or other server that cannot communicate with a browser natively. The VPN gateway then intercepts messages from the server to the client PC. The VPN gateway webifies these messages (converts them into WebPages). Unfortunately, most SSL/TLS VPN gateways can only webify a few applications.

In other cases, the SSL/TLS VPN gateway connects the client PC to a subnet of the network. The client can then connect to any host on the subnet.

- What does the client need to have? For basic operation, the client only needs to have a browser that works with SSL/TLS. It is difficult to find a computer that does not have a browser or whose browser cannot work with SSL/TLS. Consequently, SSL/TLS can work with any client PC connected to the Internet, including those at work, those in hotels, those on mobile notebook PCs, and those in Internet cafes. This makes SSL/TLS extremely attractive as a remote access VPN.
- Although being able to work with browsers alone is attractive, many SSL/TLS VPN gateways can only provide their full functionality if the user downloads an SSL/TLS VPN agent program and installs it on the client PC. Without this agent, only simple SSL/TLS access to web servers and to webified applications may be possible.

In addition, SSL/TLS is a dangerous because it normally leaves information on the client PC's hard drive after the user finishes an SSL/TLS session. This is a serious security risk for people working at public computers. Many SSL/TLS VPN agents can "erase the user's fingerprints," that is, remove all traces of their session.

The agent may also be needed for the SSL/TLS VPN gateway to verify that the client PC is up to date in its security.

Although agents provide greater functionality and security, installing an agent on the client PC requires administrative privileges. On company-owned notebooks and desktop PCs, this should not be a problem. However, in Internet cafes and other public places containing rentable PCs, the user almost never has either the permission or the administrative access to install additional software[7].

### 3. Firewall

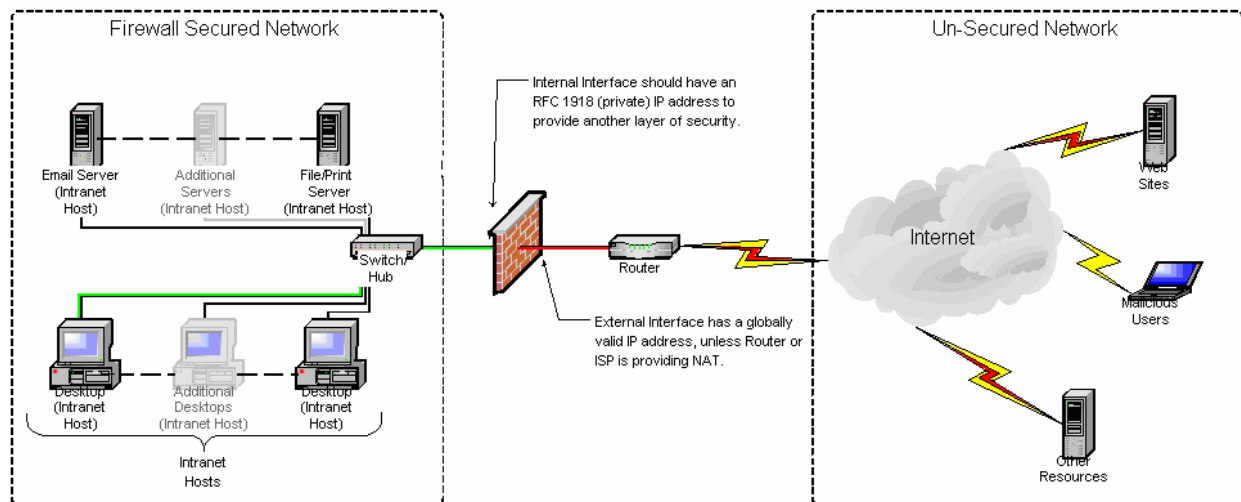
A firewall is a middleware device implemented in hardware and/or software which interconnects two Network Branches and prevent some communications forbidden by the security policy.

A firewall is designed to keep specified types of traffic from passing from the external network to the internal network. This allows administrators to control what enters the local network and keep undesirable data out. In addition to filtering this inbound traffic, a firewall can also keep specified types of traffic from passing from the internal network to the external (outbound traffic), thus preventing internal users from sending various types of data, or sending data to particular destinations.

There are different zones of trust, typically, zones of trust include the Internet (a zone with no trust) and an internal network (a zone with high trust). The ultimate goal is to provide controlled connectivity between zones of differing trust levels through the enforcement of a security policy and connectivity model based on the least privilege principle.

Among the basic types of firewalls there are several classes depending on:

- Whether the communication is being done between a single node and the network, or between two or more networks.
- Whether the communication is intercepted at the network layer, or at the application layer.
- Whether the communication state is being tracked at the firewall or not.[8]



*Fig.4: Typical firewall scenario corporate network behind a firewall*

**Personal Firewall**, a software application which normally filters traffic entering or leaving a single computer.

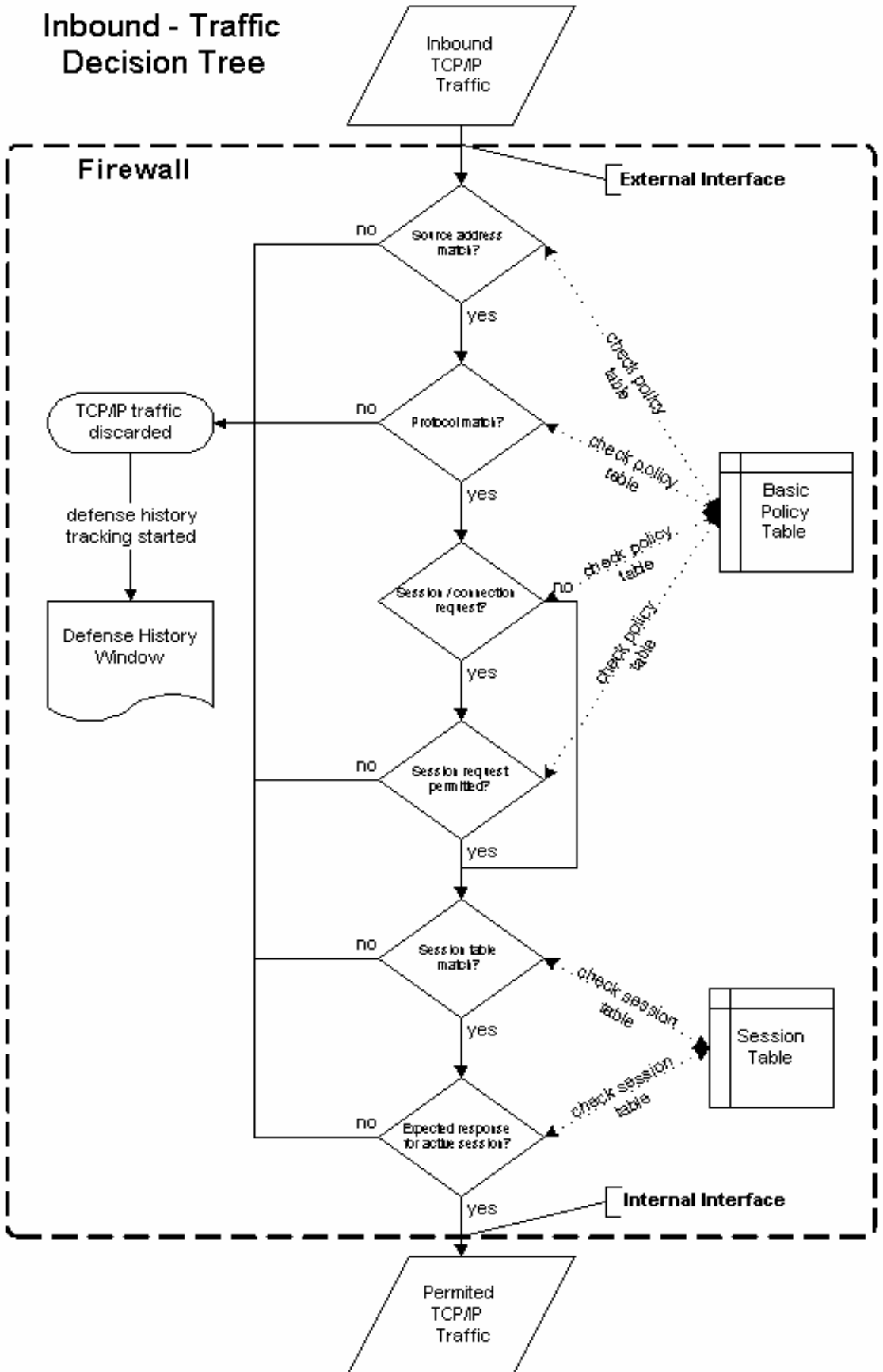
**Network firewall**, running on a dedicated device, positioned on the boundary of two or more networks.

**Application-level Filters** often called proxies, they can read the data part of each packet in order to make more intelligent decisions about the connection.

Firewalls inspect each packet and decide whether it should be allowed to pass the firewall and continue traveling towards its destination, or be discarded. But in some cases this information is not enough. The administrator of the firewall might want to allow packets to pass the firewall according to the context of the connection, and not just the packet header characteristics. This deep packet inspection provides a much finer grained control.

### Inbound - Traffic Decision Tree

In this diagram is possible to observe the typical chart flow of a firewall behavior. As a new packet arrives, it's subjected to a series of selection steps, being forwarded or discarded according to predefined rules. There are many different types of possible policies to each a set of rules are defined. If a packet succeeds in meeting all the necessary requirements, and arrives the end of the selection process, it will be forwarded to the internal interface and allowed to reach the internal network.[9]



## 4. Crossing Firewalls

### 4.1 Firewall pinhole

refers to ports that is explicitly opened through a firewall in order to allow a particular application connect to specific services in the protected network.

Allowing open holes in a firewall exposes the protected system to malicious abuse. A fully closed firewall, in spite of its ideal carma, would prevent any applications from accessing information on the other side of the firewall. In some situations, services running behind a firewall (e.g. VPN server) it is necessary to carefully open small holes in firewalls that are very small and restricted (hence the term pinhole). For best protection, the mechanism for opening the pinhole in the firewall must implement some form of validation and security that will protect the system behind the firewall.

For firewalls performing a network address translation (NAT) function, the mapping between the {external address, external port} tuple and the {internal address, internal port} tuple is often called a pinhole.

Pinholes can be created manually or programmatically. They can be temporary (created dynamically for a specific duration such as for a dynamic connection) or permanent (such as for signaling functions).

Firewalls sometimes automatically close pinholes after a period of time (typically a few minutes) to minimize the security exposure. Applications that require a pinhole to be kept open often need to generate artificial traffic through the pinhole in order to cause the firewall to restart its timer.

### 4.2 Hole punching

is a technology that allows establishment of communications between two parties, who are both behind restrictive firewalls. Both clients initiate a connection with a third-party server that uncovers external and internal address information for them. As each client initiated the request to the server, the server is aware of their IP addresses and port numbers assigned for that specific session. This information is provided to both host allowing them to establish then a direct communication between each other. Having valid port numbers causes the firewalls to accept the incoming packets from each side. UDP hole punching and TCP hole punching respectively use User Datagram and Transmission Control Protocols.

NAT traversal through UDP hole punching is a method for establishing bidirectional UDP connections between Internet hosts in private networks using NAT. It does not work with all type of NATs as their behavior is not standardized.

The basic idea is to have each host behind the NAT contact a third well-known server (usually a STUN server) in the public address space and then, once the NAT devices have established UDP state information, to switch to direct communication hoping that the NAT devices will keep the states despite the fact that packets are coming from a different host.

In order to work this technique requires a full cone NAT device. It will not work across a restricted cone NAT or a symmetric NAT.

### 4.3 UPnP

architecture allows peer-to-peer network connectivity of hosts, wireless devices, and applications. The UPnP operates in a distributed way, uses TCP/IP and HTTP to enable seamless proximity networking. It's networking architecture operates in an open structure, and is used to control data transfer among networked devices, opening for that effect ports on the middleware in between the end devices. This approach represents a serious security threat, it should never be used, it's unhappy creation was meant to home use, where the security was never particularly important.

### 4.4 SOCKS

allows client-server applications to transparently cross a network firewall. Clients behind a firewall connect to a SOCKS proxy server, that controls the eligibility of the client to access the exterior and passes the request on to the server outside, accepting then the answer from the external service and conducting it back to the client who first requested the service.[10]

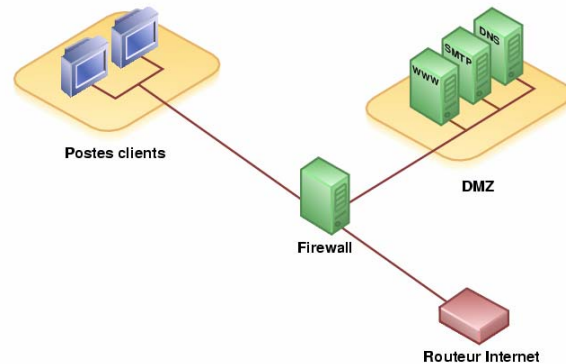
## 5. Some implementation scenarios

A common approach for an attacker is to break into a host that's vulnerable to attack, and exploit trust relationships between the vulnerable host and more interesting targets.

Possible solutions are among:

### 5.1 DMZ

Demilitarized zone (DMZ) or perimeter network is a network area that is placed between a company internal network and an external network. Connections from the internal and the external network to the DMZ are permitted, whereas connections from the DMZ are only permitted to the external network, hosts in the DMZ can't connect to the internal network. This configuration allows hosts placed in the DMZ to provide services to the exterior while protecting the internal network in case a server in the DMZ is compromised.



Networks running a number of services that have different levels of security, DMZ can be break into several "security zones". This can be done by having a number of different networks within the DMZ. For example, the access router could feed two Ethernets, both protected by ACL's, and therefore in the DMZ.

By splitting services up not only by host, but by network, and limiting the level of trust between hosts on those networks, it's possible to reduce the likelihood of a breaking on one host being used to break into the other.

Scalability of the architecture becomes easier by placing hosts on different networks. The fewer machines that there are to share the available bandwidth, the more bandwidth that each will get.

### 5.2 Running several layers of firewalls

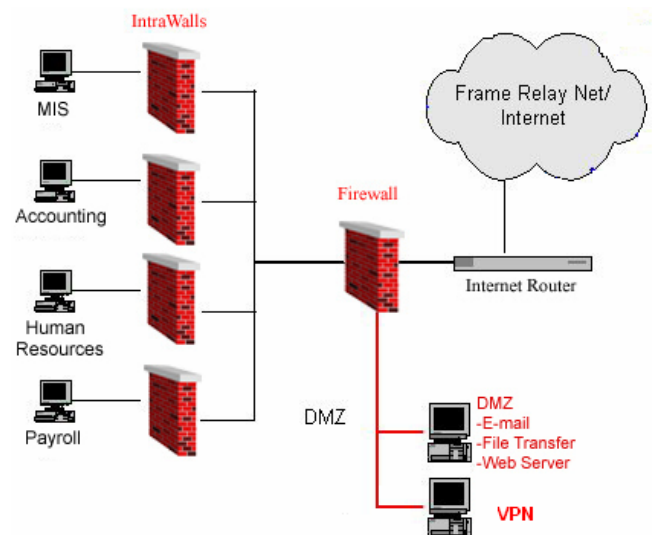
In this scenario the network is divided in several subsections behind customized and adequate firewall types.

The first layer would be a **Network firewall**, hardware type that performs NAT and packet inspection. This first firewall will create a DMZ where the services accessible from the exterior are placed, including the VPN server.

Then we have the intrawalls, normally **Application-level Filters** will be deployed, controlling the intranet access to the exterior preventing any type of traffic that may compromise security, namely VPN connections to the outside.

On the end machines **Personal Firewall** will be deployed.

These are important to prevent some type of infection, contracted in a end machine by misuse, to compromise the entire network by self propagating mechanisms, normally characteristic of malware.



The VPN access, from the exterior, is managed by the VPN server placed in the DMZ. This will be managed by the administrator allowing access to the required applications and tight controlled and updated.

## 6. Conclusions

Port range opening on the firewall, along with proprietary tunneling disable the firewall's functionality. The firewall was positioned in its location for the very reason of protecting the users and the internal network. Hence, disabling it causes a severe degradation in security.

VPN's are a direct connection from the "outside world" to the protected network environment, firewall can't block any form of malicious traffic that comes thru that tunnel, when configuring access the administrator as to think carefully what kind of resources the VPN should have access to.

Any connection established between a closed network and the outside world, represents a security risk, that as to be considered in terms acknowledgement and its respective threat. Infallible secure systems don't exist, but with the use of good sense and the adequate technology

## 7. References

- [1] VPNC, Virtual Private Network Consortium <http://www.vpnc.org/vpn-standards.html>
- [2] Security Architecture for the Internet Protocol, RFC4301, December 2005
- [3]Cisco, Deploying IPsec Virtual Private Networks, white paper, [http://www.cisco.com/en/US/products/ps6635/products\\_white\\_paper09186a0080117919.shtml](http://www.cisco.com/en/US/products/ps6635/products_white_paper09186a0080117919.shtml)
- [4] IPsec Configuration Policy Information Model, RFC 3585, August 2003
- [5] Open VPN, <http://openvpn.net/>
- [6] The Transport Layer Security (TLS) Protocol Version 1.1, RFC4346, April 2006
- [7] Transport Layer Security (TLS) Extensions, RFC3546, June 2003
- [8] Evolution of the Firewall Industry, <http://www.cisco.com/univercd/cc/td/doc/product/iaabu/centri4/user/scf4ch3.htm>
- [9] Firewalls FAQ, <http://www.faqs.org/faqs/firewalls-faq/>
- [10] Wikipedia, the Free Encyclopedia, Firewall, <http://en.wikipedia.org/wiki/Firewall>