Practical Assignment
Firewalls, Perimeter Protection, and VPNs
Washington, DC 2002 V 1.9

By Cesar Farro F.
May 27, 2003

INDICE

1. INTRODUCTION

1.1. About the company

GIAC Enterprises is an american company which main business is selling "fortune cookie sayings" by Internet. It has a provider network around the world, and its importance in the U.S.A. market is considerable because their sellings represents more than ten percent of all the Internet transactions in the U.S.A..

The GIAC's main office is in Florida. There is also a branch office located in Texas, but there is not a secure connection between them (a 256 Kbps PPP link). The main office has a 2 Mbps ADSL Internet connection.

1.2. About the business model

During the last travel of Mr. Belvedere , the GIAC's CEO, to Peru he was interested in the special skills of the people called "chamanes". These people have the power of predict the future of the others using original methods like: reading the tea, interpreting the weather, and so on. The accuracy of their predictions convinced Mr. Belvedere to set up a new business: to use Internet in order to sell these kind of predictions.

After a marketing research, done by the best professionals of GIAC Enterprises, the project was approved. The business model adopted by the company consider these main points:

- Suppliers of the predictions, this work will be the responsibility of the "chamanes"
- Partners and resellers over the world, who sell these new "product" by Internet but doing a few changes like translating the predictions in their own languages (or the languages of their markets)
- Sales people, who are employees of GIAC Enterprises with responsibilities of visit the company's clients in special days of the year.
- Teleworkers, who are employees of GIAC Enterprises but with no responsibilities of business.
- Customers, people who buy the "fortune cookie sayings" by Internet.
- The use of Internet for all processes.

It is clear that GIAC needs to invest in deploying an e-commerce infrastructure which will support this project. This work intends to offer a thecnical solution for GIAC's project, covering all the security issues needed to consider in an e-commerce business.

2. ASSIGNMENT 1 – SECURITY ARCHITECTURE

2.1. *Business Operation Model*

2.1.1. Customers

The "fortune cookie sayings" will be sold by Internet, that is why the website will consider different versions according to the appropriate language of the customer (controlled by the browser language or directly by the customer). It will be also possible to pay online using a credit card, so the web page will process e-commerce transactions using VISA, MasterCard or American Express.

The first time a customer visits the web page, and if he expresses his desire to buy a product, it will be asked to fill a form in order to capture important information of him. This process will finish sending back to him an user account and a password by e-mail. In future opportunities, these information will help him to buy the GIAC's products.

Our design consider to have the site available during the 24 hours of all the days of the year (a well known property of all Internet developments). It is also considered a first level of security: to use a secure web access ("https") in all processes that consider to get personal information of the customers or those which involve the use of an online paying method.

The web server which will be the interface between a customer and GIAC is called in our design as "web1". In the following paragraphs, specially in the diagram of the solution architecture, it will possible to identify this web server.

2.1.2. Suppliers

As we have mentioned above, our Suppliers will be those people called "chamanes". They are from Peru and are located specially in the countryside of the following states: Iquitos, Pucalpa, Piura, Arequipa and Cuzco (all of them in Peru).

In order to guarantee a secure connection between them and the main office of GIAC (located in Florida, U.S.A.), our design consider to use VPN connections. So, each provider will have a VPN client which allow them to establish a secure connection.

All of these VPN clients will use digital certificates, ensuring the process of authentication and privacy of their communications.

When a provider is asked for supplying new "message fortune sayings", he will transfer all his information through a secure connection established between his VPN client and the

VPN server located in Florida. This VPN server will give the PC's provider a valid IP address in order to allow him to access the web server of Suppliers (called in our design as "web2"). All the information will be transferred after filling special *html* forms, and will be recorded in a database server (referred in our design as "Database Server Oracle").

2.1.3. Partners

The partners will get the "message cookie sayings" sent by the Suppliers in order to translate them in their own languages (or the language of their markets). To do this, they will use VPN connections between their PC's and the web server of Suppliers ("web2").

First of all, when they establish a VPN tunnel with the VPN server, they will be allowed to access "web2" because the firewall, considered in the design, will have the appropriate rules to allow only them to access to this web server.

After they have accessed "web2", they can download the "message cookie sayings" which they want to sell. The download process also considers a MD5 digest of the message in order to help them to verify the integrity of the transfer.

As in the previous case, the authentication and privacy are guarantee because of the use of digital certificates in the VPN connections.

2.1.4. Sales People

They are people that can access to the "Service Network " in order to get the official date about products, price list, state of the warehouse, etc..

As in the previous cases, they will establish secure VPN connections with the main office. Their VPN clients will establish a VPN tunnel between theirs machines (specially lap-tops) and the VPN concentrator located in the main office.

2.1.5. Teleworkers

They are people which can work at their homes. Our design also consider that the best secure method to establish a connection between them and the main office is using VPN.

They also have a VPN client in their personal computer, but they only will be allowed to access to the "Internal Network". This will allow them to send a receive email, and to access some file servers.

2.2. Giac Enterprises Network Design

The network is designed over the criteria of "defense in depth", so our architecture is based in layers depending on the critical data.

We have segmented the network in VLANs. Each broadcast domain has an specific purpose for service and traffic. In networks where there are high traffic (because of customers, partners, or internal users) we have installed IDSs in order to mitigate the intrusion possibility.

The following diagram shows the "layers" security design:

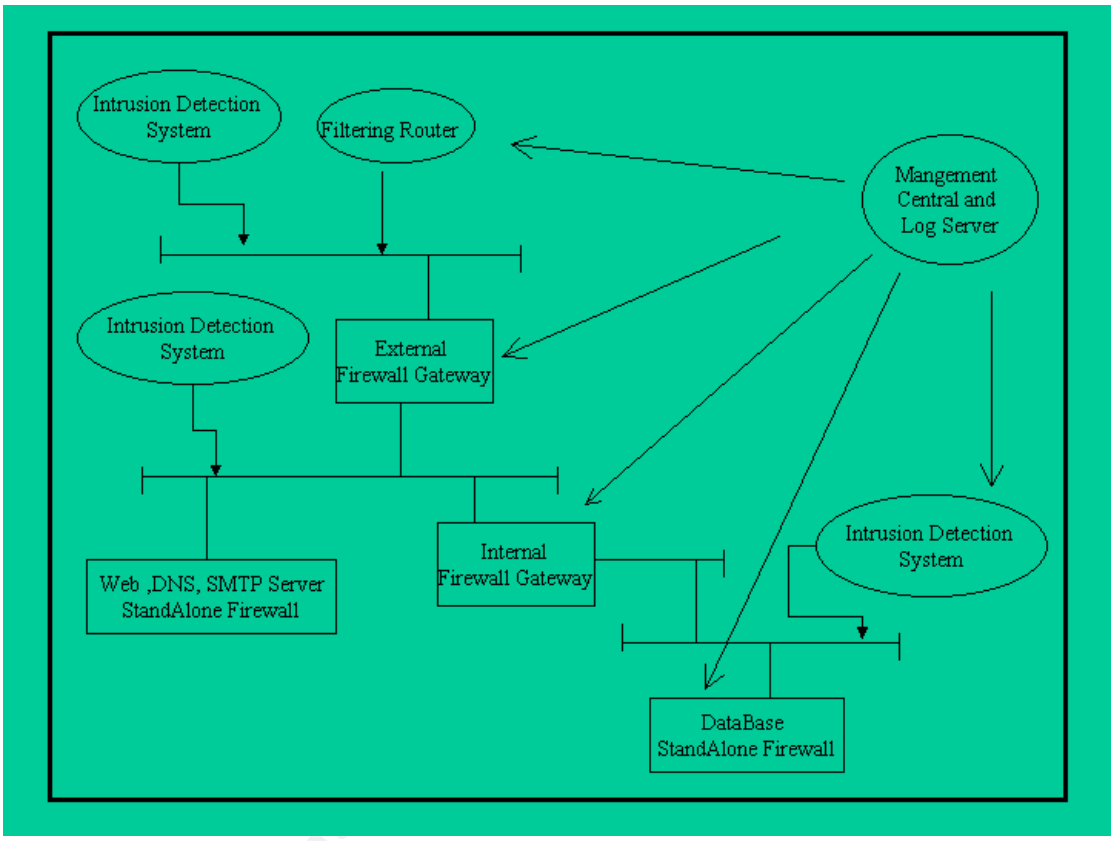


Diagram 1

This diagram shows the network, and the VLANs, in more detail:

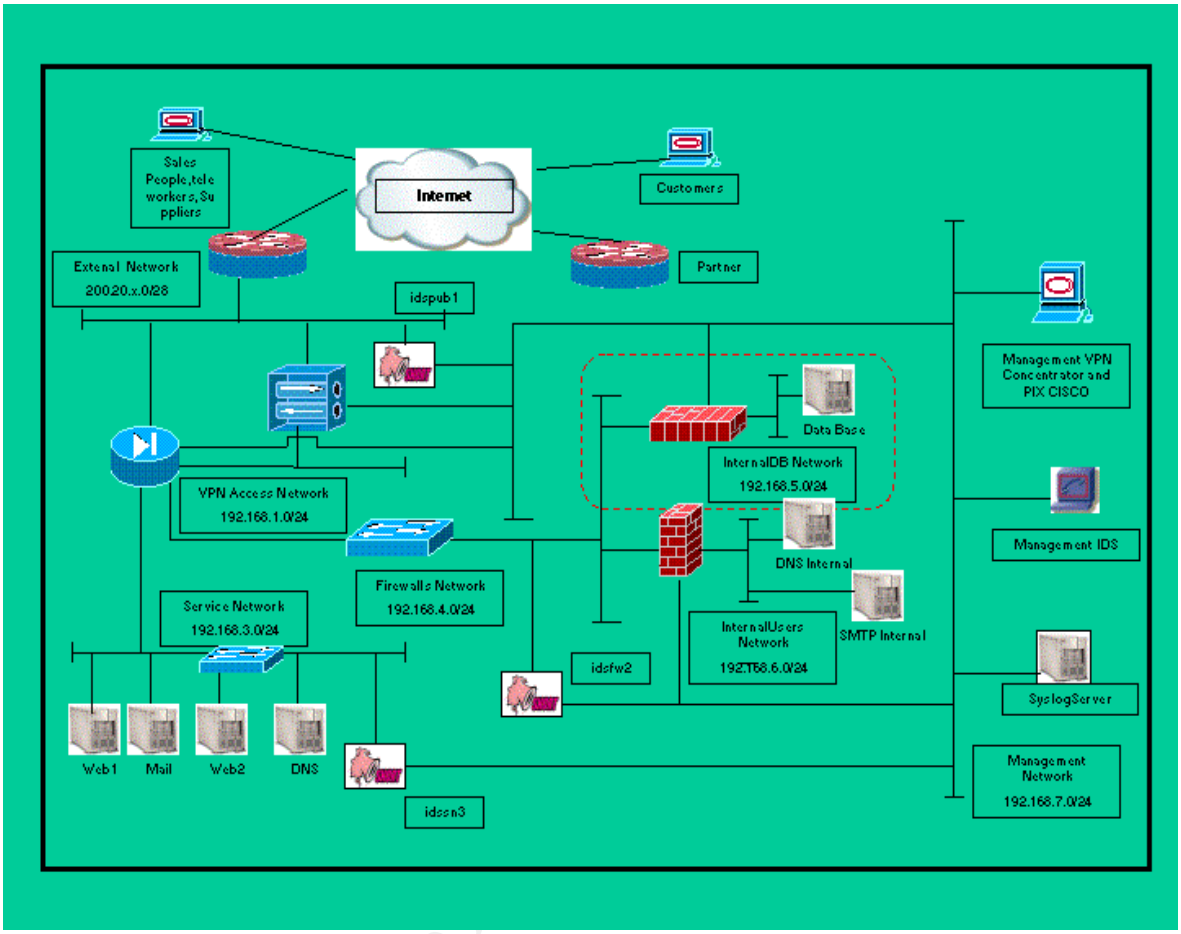


Figure 1

2.2.1. External Network

This is the public network that is freely available for everyone. However, our border router will have an special configuration in order to mitigate attacks like "denial of service" (in the part two of this work we will explain in detail its importance and configuration).

This network is directly connected to the main firewall, VPN Server, Border router and the respective IDS.

Componet	General description
1. Border Router	It is a CISCO 2600 running CISCO IOS. This model was chosen because it is a medium router.
2. External firewall	The main firewall is a CISCO PIX 525UR running firmware

	version 6.2 (2) in failover with one boxes same. It has an Ethernet card of four 10/100 ports.
3. VPN concentrator	It is a CISCO VPN3030 running OS version 3.5. It has a three cards: public, internal, and external.
4. IDS "idspub1"	It is the main IDS and it will have all the signatures of the different available services .It is an IDS Snort version 1.9.1 over Linux version 8.0.

2.2.2. VPN Access Network

It is a DMZ of the main firewall. If there are requests of the VPN clients, these requests will arrive to the public interface of the VPN server. These server will provide of valid static IP addresses to sales people, partners or teleworkers (the VPN clients). After have finished this process, they will be able to access to specific services into the internal network (the access of the appropriated service will be controlled by the main firewall).

2.2.3. Service Network

It is the main network because the main e-business server is located there. Moreover, all the public servers are also located there. For these reasons we have installed an additional IDS in this network.

The following table shows a general description of the principal components of this network:

Component	General description
Web server1	It is the main web server for clients. It will be running apache over linux 8.0, and will have a firewall standalone with iptables.
Web server2	It is the main web server for Suppliers and partners. It will be running apache over linux 8.0, and will have a firewall standalone with iptables
Mail server	It is the mail server for the GIAC's employees. It will be running sendmail over linux 8.0, and a firewall standalone with iptables.
DNS server	It is the dns server for the GIAC's employees. It will be running bind over

	linux 8.0, and a firewall standalone with iptables.
IDSsn3	It is and ids with specific rules for apache, and sendmail. It is an IDS snort version 1.9.1 over linux version 8.0.

2.2.4. *Firewall Network*

We have three firewall for different security levels. It is the network where the most advanced hackers must be detected and mitigated.
One of these firewall will control the access to our "database server", and other is controlling the Internet access.

Also, in these network we have installed an additional IDS.

2.2.5. *Internal Data Base Network*

Our main database server is located in this network. This network is the most protected area, controlling the access through two firewalls and an IDS.

If you analyze our design, you can see that if hackers access to the WebServer, it will be extremily difficult to access to our database server because of an internal firewall and an IDS.

2.2.6. *Internal users network*

It is the network where the employees are. All employee request to Internet will be first sent to the internal firewall which will be configured as a proxy server.

2.2.7. *Management Network*

Considering the several components of our security architecture, in this network it will be installed all the management workstations for firewalls, IDSs, routers, switches, etc.. Additionally, it will be installed a Syslog server .

The following table shows the IP addressing scheme which we will use:

Network	*Device (Interface)*	*IP Address*	*Purpose*
External 200.20.x.0/28	Border Router	200.20.x.1	Main Interface
	Firewallpix(Outside)	200.20.x.2	Main Interface
	Firewallpix(Outside)	200.20.x.3	Failover
	VPNServer(Public)	200.20.x.4	Main Interface

	Web Server1	200.20.x.5	NAT
	Web Server2	200.20.x.6	NAT
	Mail	200.20.x.7	NAT
	DNS	200.20.x.8	NAT
	FirewallnetfiltUsers	200.20.x.9	NAT
VPN Access 192.168.1.0/24	Firewallpix(intf0)	192.168.1.1	Main Interface
	Firewallpix(intf0)	192.168.1.2	Failover
	VPNServer(Private)	192.168.1.4	Main Interface
Firewalls 192.168.2/24	Firewallpix(intf1)	192.168.4.1	Main Interface
	Firewallpix(intf1)	192.168.4.2	Failover
	FirewallnetfilUsers(eth0)	192.168.4.4	Main Interface
	FirewallnetfilDB(eth0)	192.168.4.5	Main Interface
InternalDB 192.168.5.0/24	FirewallnetfilDB(eth1)	192.168.5.1	Main Interface
	DataBaseServer(eth0)	192.168.5.2	Main Interface
InternalUsers 192.168.6.0/24	FirewallnetfilUsers(eth1)	192.168.6.1	Main Interface
	InternalDNS (eth0)	192.168.6.2	Main Interface
	InternalMail (eth0)	192.168.6.3	Main Interface
Service 192.168.3.0/24	Firewallpix(inside)	192.168.3.1	Main Interface
	Firewallpix(inside)	192.168.3.2	Failover
	Web Server1(eth0)	192.168.3.4	Main Interface
	Web Server2(eth0)	192.168.3.5	Main Interface
	DNS(eth0)	192.168.3.6	Main Interface
	Mail(eth0)	192.168.3.7	Main Interface
Management 192.168.7.0/24	Firewallpix(intf1)	192.168.7.1	Main Interface
	Firewallpix(intf1)	192.168.7.2	Failover
	FirewallnetfilDB(eth2)	192.168.7.4	Main Interface
	FirewallnetfilUsers(eth2)	192.168.7.5	Main Interface
	Idspub1(eth1)	192.168.7.6	Main Interface
	Idsfw2(eth1)	192.168.7.8	Main Interface
	Idssn(eth1)	192.168.7.9	Main Interface

	ManagementFwPIX ConVPN	192.168.7.12	Main Interface
	ManagementIDS	192.168.7.11	Main Interface
	SyslogServer-(eth0)	192.168.7.10	Main Interface

Table 1 ***– Address Assignment***

3. ASSIGNMENT 2 – SECURITY POLICY TUTORIAL

We have a CISCO router as our border router, the first defense against outside hackers. It must be the first defense for GIAC Enterprises network, this means that we have to configure the router with the appropriate statements in order to cover all the security issues.

3.1. Border Router Configuration :

3.1.1. General Configuration

The following table shows a description of the principal commands we have used to configure this router:

Command	Description
hostname giac0001	It is important to use hostname that are not explicity related to the name of the company which it belongs to.
service password-encryption **enable secret <password>**	A good practice is to force the device to display passwords in an encrypted way.
no snmp-server	The Simple Network Monitoring Protocol can be a useful means of finding the status of devices on the network, but is vulnerable to attacks. That it is why we do not use it.
no-service tcp-small-servers **no-service udp-small-servers**	The name "small.servers" refer to those services running on TCP and UDP ports less than port 20. If minor TCP/IP servers are disabled ,access to the Echo, Discard, Chargen, and Daytime ports causes that the router sends a TCP RESET packet or an to "ICMP port unreachable" messages (UDP services) to the sender, refusing the original incoming packet.
no ip http server	This command eliminates the possibility to use HTTP to manage the router (which may be used for a denial of service attack if it is used in conjuntion with the web server).

no ip bootp server	We consider that a border router is not the best central repository for IOS configuration files.
no cdp run	Cisco Discovery Protocol is a proprietary Cisco solution which allows to directly connected devices to exchange configuration information of each of them.
no ip source-route	An attacker could use this technique to spoof the IP address of a valid host. IP source routing is a way to specify the path a packet uses to travel between hosts.
logging on **logging 192.168.7.10** **no logging console**	To enable the sending of the log messages to the logging server, without writing any log message to the console.
banner login ^C **Unauthorized access is prohibited.** **^C**	People will be warned not to try to access the system without permission.
(config-line)#line vty 0 4 **(config)#transport input ssh** **(config-line)#login authentication is -in** **(config-line)#exec-timeout 5 0** **(config-line)#end**	We have to provide a secure telnet access (VTY); our choice is for SSH connections and timeout of 5 minutes.

Table 1 – Border Router – General Configuration

3.1.2. ICMP and Broadcast traffic Control

ICMP and Broadcast traffic Control **"interface eth0/0 , interface s0/0"**	**Description**
no ip unreacheable	To block all ICMP host unreacheable

	messages.
no ip directed-broadast	Since a directed broadcast is a good method to stimulate a response from all hosts connected to the subnet, it could be used to gain information about the net.
no ip redirect	To deny all ICMP redirect messages
no ip proxy-arp	To discard any ip datagram containing this option by disabling arp

Table 2 – Border Router – ICMP Configuration

3.1.3. Ingress and Egres Filtering

An ACL (Access Control List) is a sequence of permit and deny statements used to control IP and TCP/UDP traffic across the router, with the ultimate goal to allow or block traffic to specific destinations.

In an ACL rules are evaluated exactly in the order they are listed. For this reason, it is very important to list rules from more specific to general.

ACLs belong to the following categories:

ACL Type	Function
Standard	Identified by a numeric value in the range of 1-99. Filtering is accomplished based only on source and destination IP addresses.
Extended	Identified by a numeric value in the range of 100-199. Filtering is accomplished based on source and

	destination IP addresses, as well as TCP/UDP ports.
Named	They are roughly identical to Extended ACLs but they are identified by a "name" instead of a numeric value.

Table 3 – Cisco IOS: standard ACL

We will focus on standard and extended ACLs, the ones we will use to filter incoming traffic on the router, examining their syntax and structure. Syntax for standard and extended ACLs is indicated in Table 3, 4 and 5, where command 1 (global configuration mode) refers to ACLs definition, while command 2 (interface configuration mode) refers to interface linking.

	STANDARD ACL
1	**access-list** *access-list-number* **{permit | deny}** *source source-wildcard* **[log]**
	Parameters list: • *access-list-number:* with a numerical value ranging from 1 to 99, it identifies the access list to which the current entry belongs • **permit***:* keyword specifying packets to be forwarded • **deny***:* keyword specifying packets to be rejected • *source:* IP address of host or network to be matched • *source-wildcard:* it is an optional field; corresponds to a mask specifying which bits in the IP address field are matched. It has a 1 in any position indicating a do not care bit, and a 0 in any position that is to be strictly followed. When the field is omitted, an all 0s mask is assumed. • **log***:* enables logging messages to be sent to the console when a packet entering the router matches an entry.
2	**ip access-group** *access-list-number* **{in | out}**
	Parameters list: • **in & out:** inbound or outbound direction

Table 4 – Cisco IOS: standard ACL syntax

	EXTENDED ACL FOR *TCP/UDP* PROTOCOLS

1	**access-list** *access-list-number* **{permit | deny}** **{***protocol* **|** *protocol-keyword***}** *source source-wildcard* **[***operator source-port* **|** *source-port***]** *destination destination-wildcard* **[***operator destination-port* **|** *destination-port***] [established] [log]**
	Parameters list: • *access-list-number:* with a numerical value ranging from 100 to 199, it identifies the access list to which the current entry belongs • **permit***:* keyword specifying packets to be forwarded • **deny***:* keyword specifying packets to be rejected • *protocol*: TCP, UDP • *source* **&** *destination:* IP address of host or network to be matched • *source-wildcard* **&** *destination-wildcard:* it is an optional field; corresponds to a mask specifying which bits in the IP address field are matched. It has a 1 in any position indicating a do not care bit, and a 0 in any position that is to be strictly followed. When the field is omitted, an all 0s mask is assumed. • *operator:* in this is a qualifying condition (a math operator line eq, gt, lt, …) • *source-port* & *destination-port*: a decimal value ranging from 0 to 65535 indicating a TCP or UDP port. • **established**: this keyword is optional; when used it enforces a control with respect to ACK and RST bit within a TCP packet to verify if the packet is part of a previously established connection or not (obviously, it cannot be used for UDP protocol). • **log***:* enables logging messages to be sent to the console when a packet entering the router matches an entry.
2	**ip access-group** *access-list-number* **{in | out}**
	SEE TABLE 3

Table 5 – Cisco IOS: extended ACL syntax for TCP and UDP

	EXTENDED ACL FOR *ICMP* PROTOCOL
1	**access-list** *access-list-number* **{permit | deny} icmp** *source source-wildcard destination destination-wildcard* **[***icmp-type* **[***icmp-code***] |** *icmp-message***]**
	Parameters list for an extended ICMP ACL are the same used for TCP and UDPO traffic, except from the following parameters: • *icmp-type:* number between 0-255 • *icmp-code:* number between 0-255

	• *icmp-message* (i.e. echo-reply, port unreachable, …) which identify exactly what their respective names say.
2	**ip access-group** *access-list-number* **{in | out}**
	SEE TABLE 3

Table 6 - Cisco IOS: extended ACL syntax for ICMP

3.1.4. The Internet Interface – Applied on Serial 0/0 Inbound

```
!  Blocking all traffic with source IP address corresponding to Our address assignment.
access list  101 deny ip 200.20.x.0 0.0.0.240  log

! Blocking  multicast and broadcast traffic , plus loopback address
access-list 101 deny ip 224.0.0.0 31.255.255.255 log
access-list 101 deny ip 255.0.0.0 0.255.255.255 log
access-list 101 deny ip 127.0.0.0 1  0.255.255.255 log

!Blocking all traffic with source IP address in the private  range (RFC 1918 1)  to avoid
IP !spoofing.
access-list 101 deny ip  10.0.0.0 0.255.255.255 log
access-list 101 deny ip  172.16.0.0 1.15.255.255 log
access-list 101 deny ip  192.168.0.0 0.0.255.255 log

!Blocking packets with no IP address
access-list 101 deny ip  host 0.0.0.0  any log

!Allowing SSH Connections from the outside to Serial 0/0 intf
access-list 101 permit tcp any  host 200.20.x.1 eq 22 log

!Allowing  HTTP /HTTPS  traffic to the Web Server of Customers
access-list 101 permit tcp any host 200.20.x.5 eq www  log
access-list 101 permit tcp any host 200.20.x.5 eq 443 log

!Allowing  HTTP /HTTPS  traffic to the Web Server of  Suppliers and Partners.
```

[1] http://www.fax.org/rfsc/rfc1918.html

```
access-list 101 permit tcp any host 200.20.x.6 eq www log
access-list 101 permit tcp any host 200.20.x.6 eq 443 log

!Allowing  SMTP   traffic to the Mail Server.
access-list 101 permit tcp any host 200.20.x.8 eq smtp  log
!Allowing  DNS   traffic to the DNS Server.
access-list 101 permit udp any host 200.20.x.8 eq domain  log
access-list 101 permit tcp any host 200.20.x.8 eq domain  log

!Allowing  VPN    traffic to the VPN Server.
access-list 101 permit tcp any host 200.20.x.4 eq 500  log
access-list 101 permit 50 any host 200.20.x.4 log
access-list 101 permit 51 any host 200.20.x.4 log

!Allowing only established traffic to protect our Company from DOS attacks
access-list 101 permit tcp any 200.20.x.0 0.0.0.240 established
access-list 101 permit tcp any any  established

!Deny traffic that comes from address reserved by IANA [2], the internet Assigned
Numbers !Authority
access-list 101 deny 0.0.0.0 0.255.255.255 log
access-list 101 deny 1.0.0.0 0.255.255.255 log
access-list 101 deny 2.0.0.0 0.255.255.255 log
access-list 101 deny 3.0.0.0 0.255.255.255 log
access-list 101 deny 4.0.0.0 0.255.255.255 log
access-list 101 deny 5.0.0.0 0.255.255.255 log
access-list 101 deny 6.0.0.0 0.255.255.255 log
access-list 101 deny 6.0.0.0 0.255.255.255 log
access-list 101 deny 6.0.0.0 0.255.255.255 log
access-list 101 deny 6.0.0.0 0.255.255.255 log
access-list 101 deny 7.0.0.0 0.255.255.255 log
```

[2] This list is updated periodically, and you can find the latest list at the following URL : http://www.iana.org/assigments/ipv4-address-space

[3] http://www.itc.virginia.edu/desktop/security/local_summary.html

```
access-list 101 deny 8.0.0.0 0.255.255.255 log
access-list 101 deny 9.0.0.0 0.255.255.255 log
access-list 101 deny 10.0.0.0 0.255.255.255 log
access-list 101 deny 11.0.0.0 0.255.255.255 log
access-list 101 deny 12.0.0.0 0.255.255.255 log
access-list 101 deny 13.0.0.0 0.255.255.255 log
access-list 101 deny 14.0.0.0 0.255.255.255 log
access-list 101 deny 15.0.0.0 0.255.255.255 log
access-list 101 deny 16.0.0.0 0.255.255.255 log
access-list 101 deny 17.0.0.0 0.255.255.255 log
access-list 101 deny 18.0.0.0 0.255.255.255 log
access-list 101 deny 19.0.0.0 0.255.255.255 log
access-list 101 deny 20.0.0.0 0.255.255.255 log
access-list 101 deny 21.0.0.0 0.255.255.255 log
access-list 101 deny 22.0.0.0 0.255.255.255 log
access-list 101 deny 23.0.0.0 0.255.255.255 log
.
.
.
access-list 101 deny 255.0.0.0 0.255.255.255 log

!We will deny  access to the ports that are most frequently probed by hackers, but which we ! do not have listening services on :
!Logging Services – telnet (23/tcp), SSH (22/tcp), FTP (21/tcp)
access-list 101 deny tcp any any range 21 23 log
!Small Services – time (37/tcp) -  (37/udp)
access-list 101 deny tcp any any eq 37 log
access-list 101 deny udp any any eq 37 log

!Blocking NetBIOS Traffic – 135(tcp and udp), 137 (udp), 138 (udp), 139(tcp), Windows !2000 –earlier ports plus 445 (tcp and udp)
access-list 101 deny tcp  any any range  135   139   log
access-list 101 deny udp any any range   135  139  log
!Miscelaneous – TFTP (69/udp), finger (79/ tcp), syslog (514/udp)
access-list 101 deny udp any any eq 69 log
access-list 101 deny tcp any any eq 79 log
access-list 101 deny udp any any eq 514 log
```

```
!Blocking  NNTP News Server Traffic
access-list 101 deny tcp  any any eq 119  log

! SNMP (161/tcp, 161/udp, 162/tcp and 162/udp)
access-list 101 deny tcp  any any range 161 162 log
access-list 101 deny udp any any range 161 162 log

! SOCKS !(1080/tcp) [3]
access-list 101 deny udp any any eq 1080 log

!Bloking X- Windows Trojan
access-list 101 deny tcp any any range 6000 6255  log

! According to SANS TOP TEN RECOMMENDATIONS
access-list 101 deny tcp any any eq sunrpc log
access-list 101 deny udp any any eq sunrpc log
access-list 101 deny tcp any any eq  2049 log
access-list 101 deny udp any any eq  2049 log
access-list 101 deny tcp any any eq  4045 log
access-list 101 deny udp any any eq  4045 log

! Blocking Known Trojan
access-list 101 deny udp any any eq  34555 log
access-list 101 deny udp any any eq  27573 log
access-list 101 deny udp any any eq  27444 log
access-list 101 deny udp any any eq  27374 log

! Blocking and logging all other packets
access-list 101 deny udp any any  log

!Applying  ACLS
ip access-group  101 in serial0/0
```

Table 7 - The Internet Interface – Applied on Serial 0/0 Inbound

3.1.5. The Intranet Interface – Applied on Ethernet 0/0 inbound

```
! Allowing outbound traffic from public IP address only
access-list 102 permit ip 200.20.20.0 0.0.0.240 any log
access-list 102 deny ip any any log
!Apply ACLS
ip accesss-group 102 in ethernet0/0
```

Table 8 - The Intranet Interface – Applied on Ethernet 0/0 inbound

3.2. Second line of defense : firewalls security policy

We are using two CISCO PIX 525 equipments, version 6.2(2), working in failover mode. These devices have, each of them, one network interface of four ports (our design only uses six ports in total).

For the failover process we are using an interface called "sfa". This interface allows communicate in a faster way, enabling that the all the connections that are established through one firewall can be recovered.

The set of IP addresses, security levels, and the names that we use for the interfaces are listed in the following table:

Interface Name	*IP Address*	*Netmask*	*Security level*	*Interface Name*
Ethernet0	200.20.x.2	255.255.255.240	s0	Outside
Ethernet1	192.168.4.1	255.255.255.0	s100	Inside
Ethernet2	192.168.3.1	255.255.255.0	s5	dmz_ser
Ethernet3				
Ethernet4	192.168.1.1	255.255.255.0	s10	dmz_vpn
Ethernet5	192.168.7.1	255.255.255.0	s25	Mgmt
Ethernet6	11.1.1.1	255.255.255.0	s55	link failover

Table 01 – Configuration of Interfaces

Principal commands used during the configuration	
Command	**Description**
nameif ethernet0 outside security0 nameif ethernet1 inside security100 nameif ethernet2 dmz_ser security5 nameif ethernet4 dmz_vpn security10 nameif ethernet5 mgmt security25 nameif ethernet6 sfa security55	To assign a name to each Ethernet interface and to define the security level of the interface.
interface ethernet0 100full interface ethernet1 100full interface ethernet2 100full interface ethernet4 100full interface ethernet5 100full interface ethernet6 100full	To set the speed and type of operation of each interface.
enable password 2KFQnbNIdI.2KYOU encrypted passwd 2KFQnbNIdI.2KYOU encrypted	To set an encrypted password for the configuration
hostname fwtierra domain-name ciscopix.com	Hostname of the firewall
fixup protocol ftp 21 **fixup protocol http 80** no fixup protocol h323 h225 1720 no fixup protocol h323 ras 1718-1719 no fixup protocol ils 389 no fixup protocol rsh 514 no fixup protocol rtsp 554 no fixup protocol smtp 25 **fixup protocol sqlnet 1521** no fixup protocol sip 5060 no fixup protocol skinny 2000	To enable application inspection for the protocols specified that we will be using in our network.
ip address outside 200.20.x.2 255.255.255.0 ip address inside 192.168.4.1 255.255.255.0 ip address dmz_ser 192.168.3.1 255.255.255.0 ip address dmz_vpn 192.168.1.1 255.255.255.0 ip address mgmt 192.168.7.1 255.255.255.0 ip address sfa 11.1.1.1 255.255.255.0	To assign the IP address to each interface.

failover failover timeout 0:00:00 failover poll 15 failover ip address outside 200.20.x.3 failover ip address inside 192.168.4.3 failover ip address dmz_ser 192.168.3.3 failover ip address dmz_vpn 192.168.1.3 failover ip address dmz_ges 192.168.7.3 failover ip address sfa 11.1.1.2 failover link sfa	To assign IP address to each interface to failover.
floodguard enable	Enable the "flood defender" to protect against flood attacks.
logging on logging timestamp logging console debugging logging buffered debugging logging trap debugging logging history debugging logging facility 5 logging host mgmt 192.168.7.10	To set the log in the console and the traffic syslog between the PIX and the Syslog Server.
no snmp-server location no snmp-server contact no snmp-server enable traps	We are not using snmp server for the PIX.
route outside 0.0.0.0 0.0.0.0 200.20.x.1 1 route inside 192.168.5.0 255.255.255.0 192.168.4.4 route inside 192.168.6.0 255.255.255.0 192.168.4.5	To define static routes for the internal users network and the internal database network.
ssh 192.168.7.12 255.255.255.255 ssh 192.168.7.11 255.255.255.255 ssh timeout 5	To set SSH access to the firewall to the technical Staff Management workstations.
global (outside) 1 200.20.x.9 nat (inside) 1 192.168.6.0 255.255.255.0 0 0	NAT to the Users Network go to internet.
access-list 104 permit tcp 192.168.6.0 any eq 80 access-list 104 permit tcp 192.168.6.0 any eq 8080 access-list 104 permit tcp 192.168.6.0 any eq 21	To allow the internal Users Network go to internet.

access-list 104 permit tcp 192.168.6.0 any eq 25 access-group 104 in interface inside	
static (dmz_ser, outside) 200.20.x.5 192.168.3.4 netmask 255.255.255.255 0 0	Mapping web server for customers to internet.
static (dmz_ser, outside) 200.20.x.6 192.168.3.5 netmask 255.255.255.255 0 0	Mapping web server for partners to internet.
static (dmz_ser, outside) 200.20.x.7 192.168.3.6 netmask 255.255.255.255 0 0	Mapping external mail to internet.
static (dmz_ser, outside) 200.20.x.8 192.168.3.7 netmask 255.255.255.255 0 0	Mapping external dns to internet.
access-list 101 permit tcp any host 200.20.x.5 eq 80 access-list 101 permit tcp any host 200.20.x.5 eq 8080	To allow access to our web server for customers using http, https
access-list 101 permit tcp any host 200.20.x.6 eq 80 access-list 101 permit tcp any host 200.20.x.6 eq 8080	To allow access to our web server for partners and suppliers using http, https
access-list 101 permit tcp any host 200.20.x.7 eq 25 access-list 101 permit udp any host 200.20.x.8 eq 53 access-list 101 permit tcp any host 200.20.x.8 eq 53 access-group 101 in interface outside	To allow access to our mail, dns server from internet.
static (inside, dmz_ser) 192.168.5.2 192.168.5.2 netmask 255.255.255.255 0 0 access-list 102 permit tcp host 192.168.3.4 host 192.168.5.2 eq 1521 access-list 102 permit tcp host 192.168.3.5 host 192.168.5.2 eq 1521 access-group 102 in interface dmz_ser	To allow access to our database server from web of customers and web of partners and suppliers.
static (inside, dmz_ser) 192.168.6.2 192.168.6.2 netmask 255.255.255.255 0 0 static (inside, dmz_ser) 192.168.6.3 192.168.6.3 netmask 255.255.255.255 0 0 access-list 102 permit tcp host 192.168.3.7 host 192.168.6.2 eq 53	To communicate the internal mail ,dns server and external mail ,dns server.

access-list 102 permit tcp host 192.168.3.6 host 192.168.6.3 eq smtp	
static (mgmt, dmz_ser) 192.168.7.10 192.168.7.10 netmask 255.255.255.255 0 0 access-list 102 permit udp host 192.168.3.4 host 192.168.7.10 eq 514 access-list 102 permit udp host 192.168.3.5 host 192.168.7.10 eq 514 access-list 102 permit udp host 192.168.3.6 host 192.168.7.10 eq 514 access-list 102 permit udp host 192.168.3.7 host 192.168.7.10 eq 514	To enable the web of customers, web of partners and external mail and dns send their log to syslog server.
static (mgmt, dmz_vpn) 192.168.7.10 192.168.7.10 netmask 255.255.255.255 access-list 102 permit udp host 192.168.1.4 host 192.168.7.10 eq 514	To enable the vpn concentrator sends its log to syslog server.
static (dmz_ser, dmz_vpn) 192.168.3.5 192.168.3.5 netmask 255.255.255.255 access-list 102 permit tcp host 192.168.1.4 host 192.168.3.5 eq 80 access-list 102 permit tcp host 192.168.1.4 host 192.168.3.5 eq 8080 access-list 102 permit tcp host 192.168.1.5 host 192.168.3.5 eq 80 access-list 102 permit tcp host 192.168.1.5 host 192.168.3.5 eq 8080 access-list 102 permit tcp host 192.168.1.6 host 192.168.3.5 eq 80 access-list 102 permit tcp host 192.168.1.6 host 192.168.3.5 eq 8080 access-list 102 permit tcp host 192.168.1.7 host 192.168.3.5 eq 80 access-list 102 permit tcp host 192.168.1.7 host 192.168.3.5 eq 8080 access-list 102 permit tcp host 192.168.1.8 host 192.168.3.5 eq 80 access-list 102 permit tcp host 192.168.1.8 host 192.168.3.5 eq 8080	VPN Client: Suppliers and Partners must access to only the web server of suppliers and partners. The range of suppliers and partners are : “192.168.1.4-192.168.1.8”

static (dmz_ser, dmz_vpn) 192.168.3.4 192.168.3.4 netmask 255.255.255.255 static (dmz_ser, dmz_vpn) 192.168.3.5 192.168.3.5 netmask 255.255.255.255 static (dmz_ser, dmz_vpn) 192.168.3.6 192.168.3.6 netmask 255.255.255.255 access-list 102 permit tcp host 192.168.1.14 host 192.168.3.5 eq 80 access-list 102 permit tcp host 192.168.1.14 host 192.168.3.5 eq 8080 access-list 102 permit tcp host 192.168.1.15 host 192.168.3.5 eq 80 access-list 102 permit tcp host 192.168.1.16 host 192.168.3.5 eq 80 access-list 102 permit tcp host 192.168.1.16 host 192.168.3.5 eq 8080 access-list 102 permit tcp host 192.168.1.17 host 192.168.3.5 eq 80 access-list 102 permit tcp host 192.168.1.17 host 192.168.3.5 eq 8080 access-list 102 permit tcp host 192.168.1.18 host 192.168.3.5 eq 80 access-list 102 permit tcp host 192.168.1.18 host 192.168.3.5 eq 8080	VPN Client: Sales People must access to web server of suppliers, web partners and external mail. The range of sales people is : “192.168.1.14-192.168.1.18”
static (dmz_vpn,inside) 192.168.4.24 192.168.1.24 netmask 255.255.255.255 static (dmz_vpn,inside) 192.168.4.25 192.168.1.25 netmask 255.255.255.255 static (dmz_vpn,inside) 192.168.4.26 192.168.1.26 netmask 255.255.255.255 static (dmz_vpn,inside) 192.168.4.27 192.168.1.27 netmask 255.255.255.255 static (dmz_vpn,inside) 192.168.4.28 192.168.1.28 netmask 255.255.255.255	VPN Client: Teleworkers must access to the inside network then the internal firewall must applied the specific rules. The range of teleworker is : “192.168.1.24-192.168.1.28”

Table 02 – Configuration of the Firewall

3.3. VPN Server Security Policy

Partners, Suppliers, Salespeople and Teleworkers will connect to GIAC network from the outside world creating a VPN tunnel toward Cisco VPN Concentrator 3030. This is a modular mid-sized gateway ,allowing about 1,500 simultaneous users with a global throughput of 50 Mbps.

3.3.1. Ipsec Technical

We have chosen to use to IP Security Protocol, also known as IPSec.

3.3.2. Key Exchange

We have decided to use digital certificates X.509 because partners, suppliers, salespeople, and teleworkers are mobile clients.

Our Certificate Authority is installed over MS Windows 2K/SP3.

Our solution allows us to know which client connects to our vpn server and identify the owner of a digital certificate. Also, using Digital Certificates we are covering the **"No refuse"** issue.

3.3.3. Why use Authentication Header -AH vs. Encapsulation Security Payload – ESP ?

ESP provides data confidentially by encrypting the payload, or the data portion, of the data. It provides limited authentication compared to AH because it does not authenticate the packet´s headers, as AH does.

AH ensures data authentication by adding authentication information to the packet´s header, but does not provide any protection against the packet being read the contents of the packet are not encrypted. Also, AH does not work with NAT.

So we have decided to use ESP because we believe that our data must be sent totally encrypted over Internet. ESP is protocol 50, and also we are using :

1.- Authentication Algorithm : **ESP/MD5/HMAC-128** (ESP using HMAC Hashed Messages Authentication Coding) with the MD5 hash function using a 128 bit key, we have chosen these option because the option of 160 bits key requires more processing)

2.- Encryption Algorithm : **3DES-168**

3.- Encapsulation Mode : **Tunnel**

3.3.4. Configuration of the VPN server

We generated a digital certificate where the OU- Organization Unit is the specific group to our clients, for example the group suppliers is to all suppliers. We in the configuration have installed a Digital Certificate of VPN Server but in OU is the group “group01” as illustration.

Our configuration have divided in 7steps :

1.- Generation of Digital Certificate to VPN server.

2.- Installation of Digital Certificate of VPN server and Digital Certificate of root in the VPN Server.

3.- Activation of IKE Proporsal.

4.- Enabling to SA can work with the specific Digital certificate generated in the step 1.

5.-Creation of Group and User.

6.- Generation of Digital Certificate to Client VPN and Installation in the SoftwareVPN Client.

7.- Test with the VPN client and the VPN server.

8.- Enabling to our Syslog Server.

Step 1 : First we have generated the Digital Certificate to the VPN server , via Enrollment :

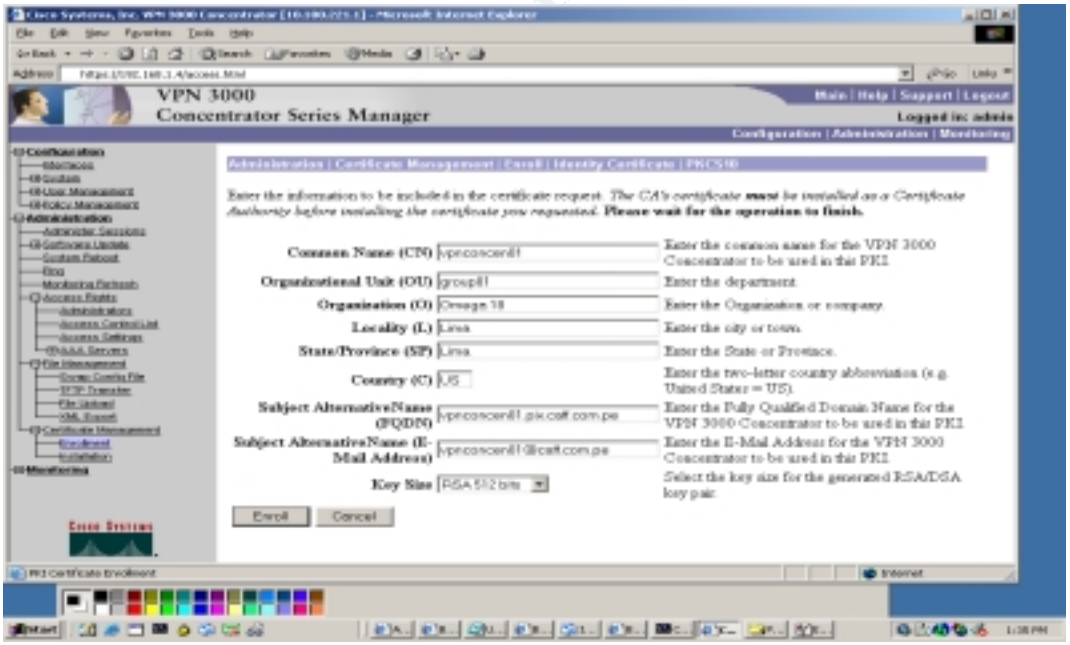


Figure 1 –Generation of Digital Certificate of VPN Server

Then we have the respective **“pksx.txt”** to finally generated manually the indicate certificate to vpn server. It´s important to say that the “OU” Organization Unit must be the same that the name group created in the vpn server.

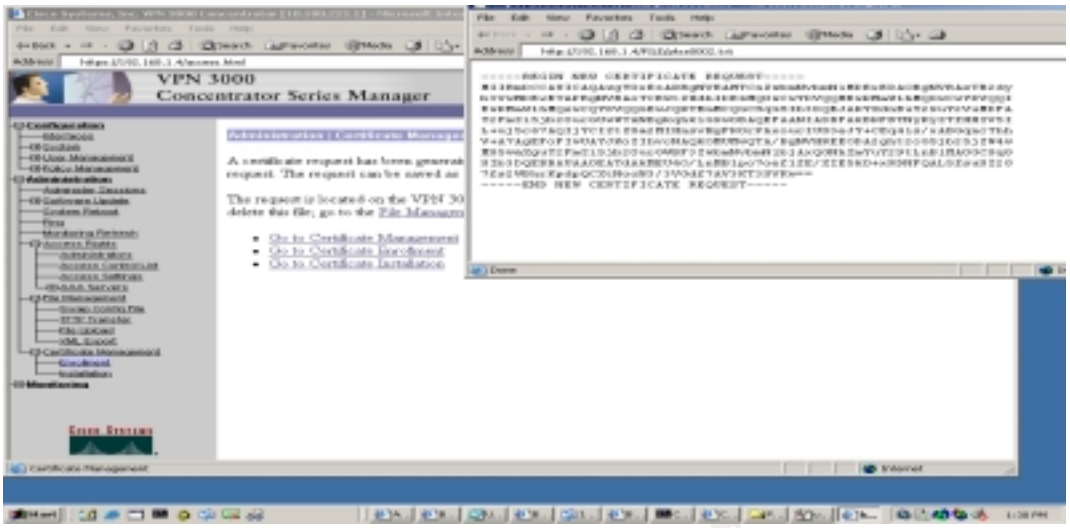


Figure 2- pks of the VPN Server

Then, with a Certificate Authority server over Windows 2000 SP3 , we have obtained the root certificate and certificate of the vpn server.

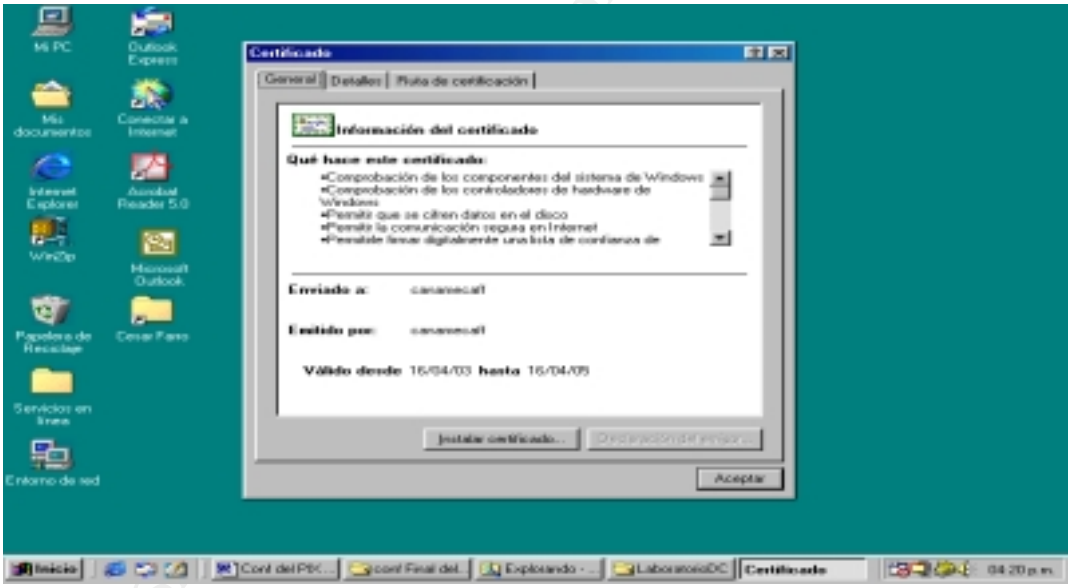


Figure 3 – Digital Certificate of the root server

The following snapshot shows the certificate of the vpn server.

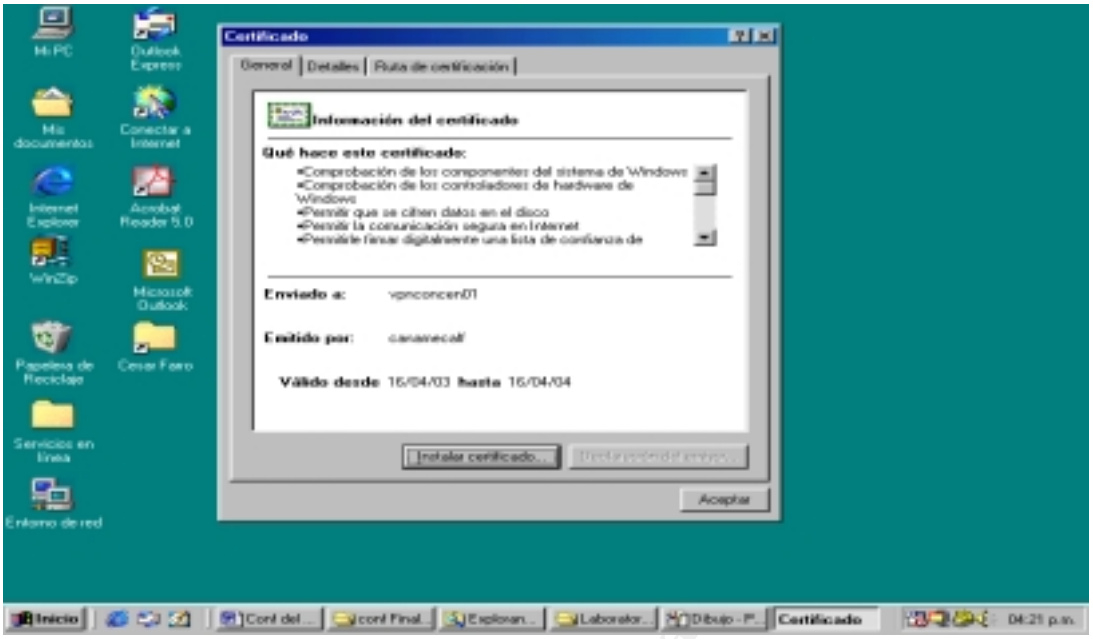
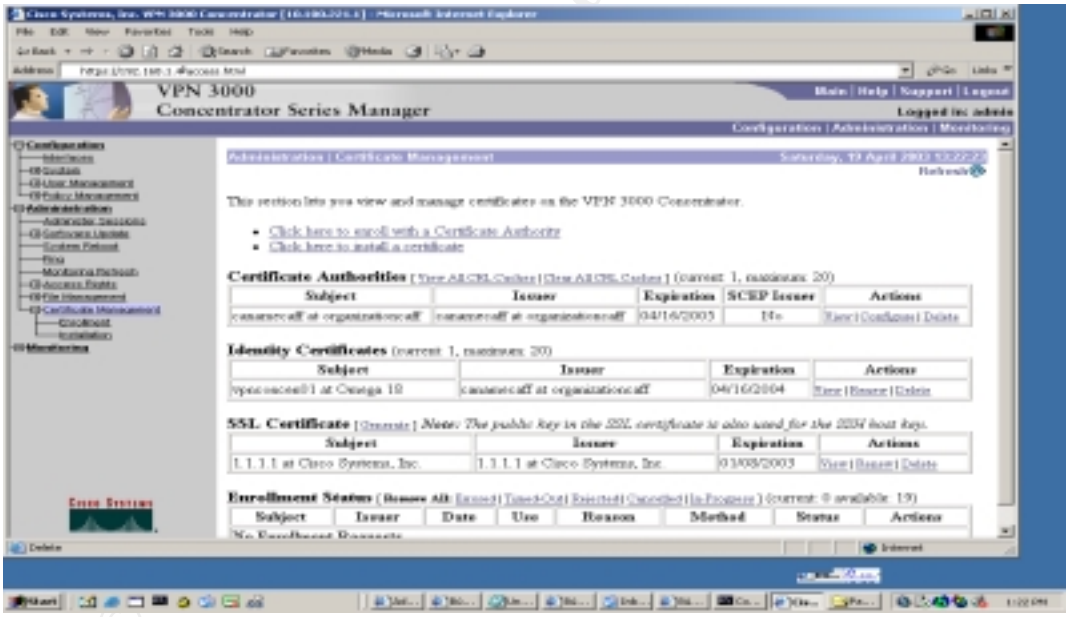


Figure 4- **Digital Certificate of the VPN Server**

Step 2 : Then we have installed both certificates in the vpn server in the menu of Certificate Manager:



Step 3 : Then We are activating the IKE Proporsal. In this case we are using **"CiscoVPNClient-3DES-MD5-RSA"**

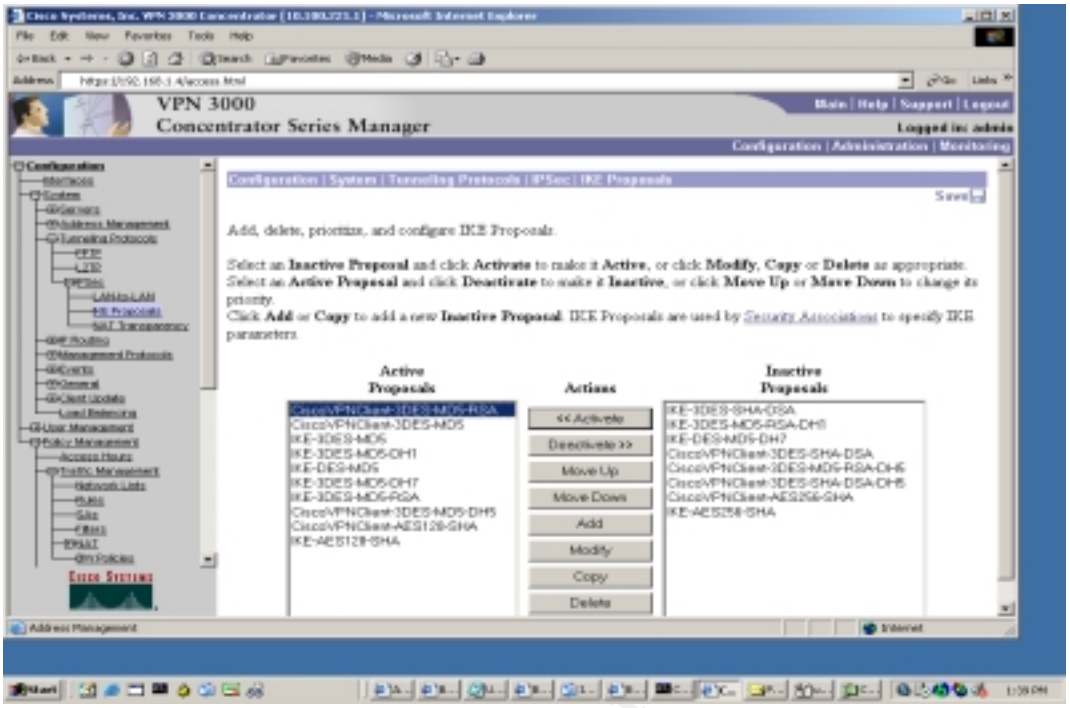


Figure 6- Active Proporsal

The following snapshot shows details about the specific "IKE Proporsal"

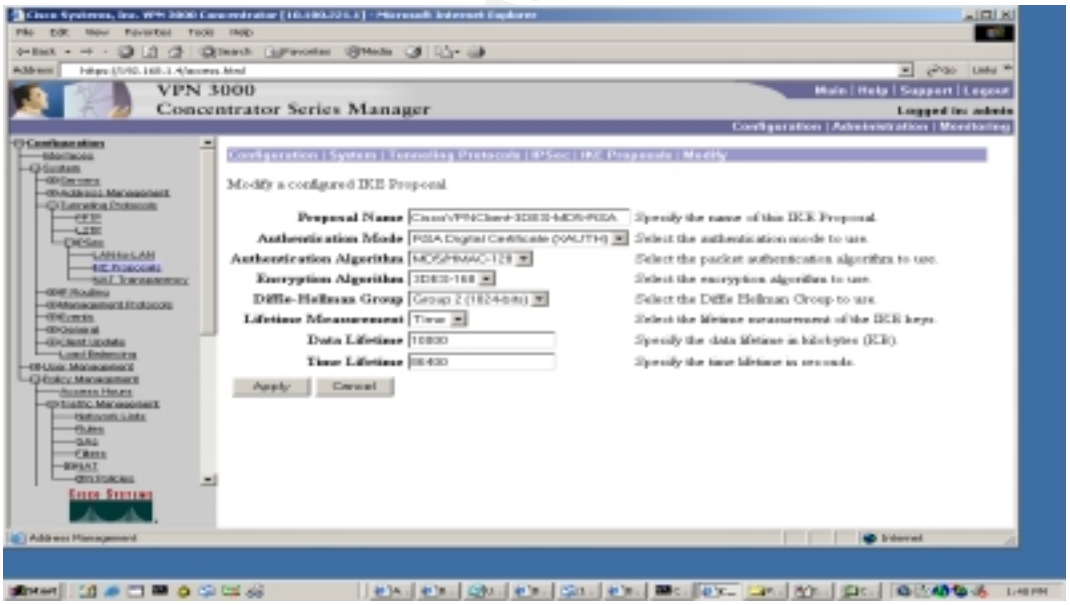


Figure 7 –Edition of IKE Proporsal

Step 4 :. We are modifying the security associations **"SA- ESP-3DES-MD5" ,** so it can work with the digital certificate that we have generated in the step 1.

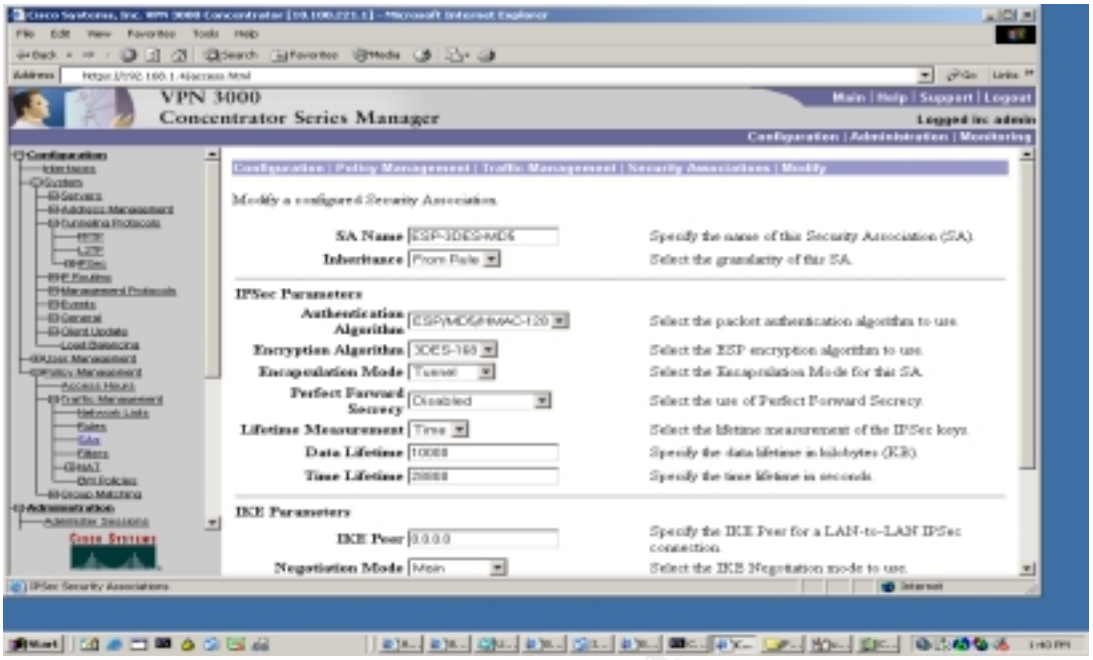


Figure 8 –Edition of IP Security Associations

In the option of IKE Parameters in Digital Certificate we have chosen “vpnconcen01” just the name of the certificate that we have generated in the step1 and in the option IKE proporsal we have chosen our specific IKE Proporsal.

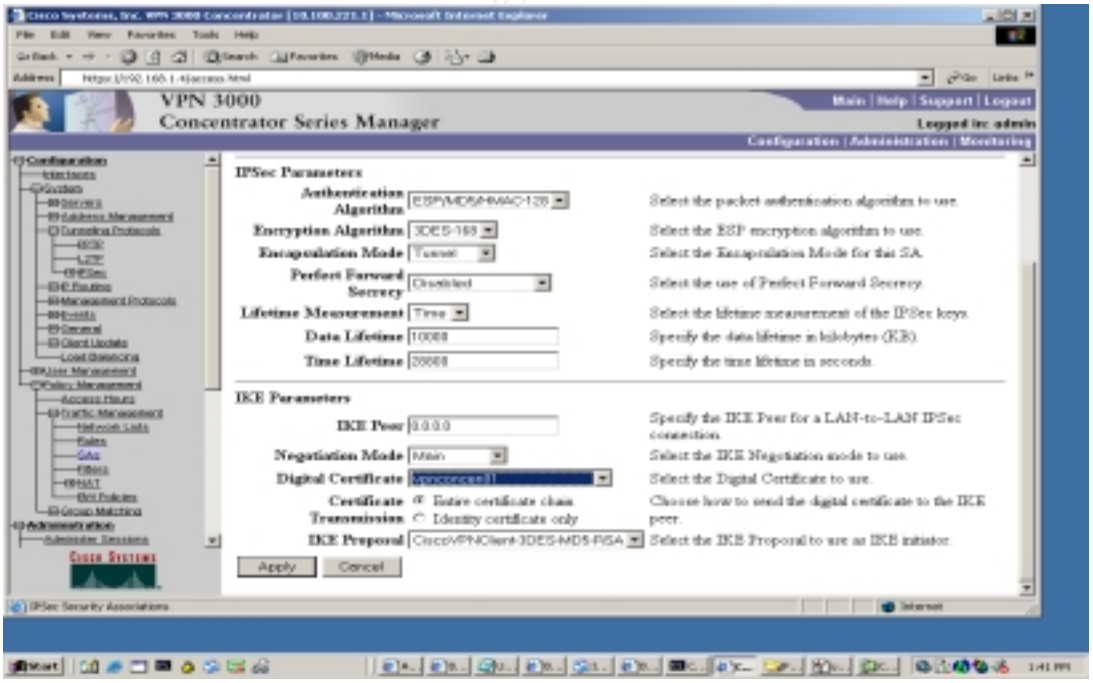


Figure 9- Edition of IP Security Associations

Step 5 : We have created a new group in these case is general , but it´s help us to can understand each group. Now we are in the identity of the group name **“group01”**

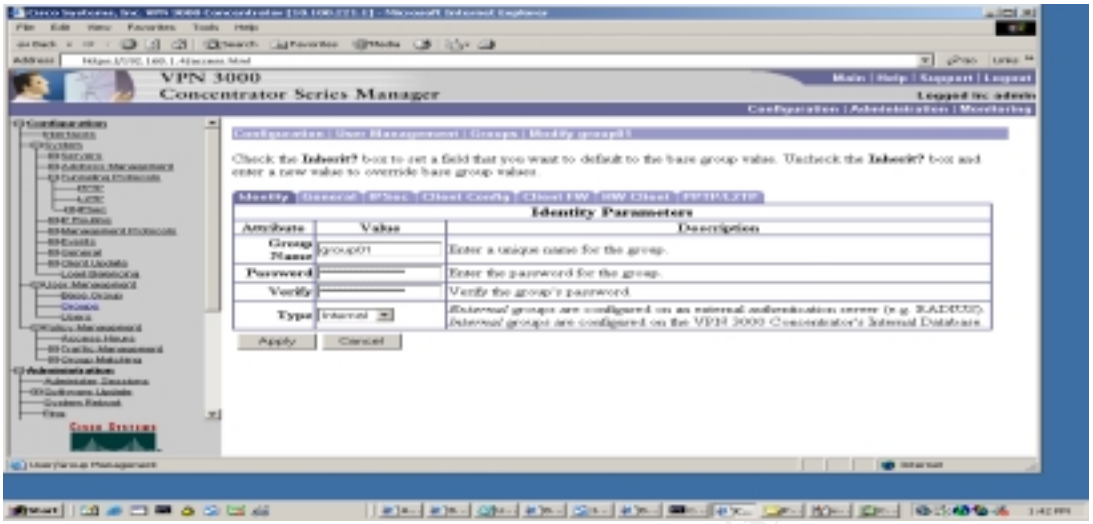


Figure 10 - Creation of the Group

Then in the properties General of the group, we have filled the IP address of the ISP ´DNS server.

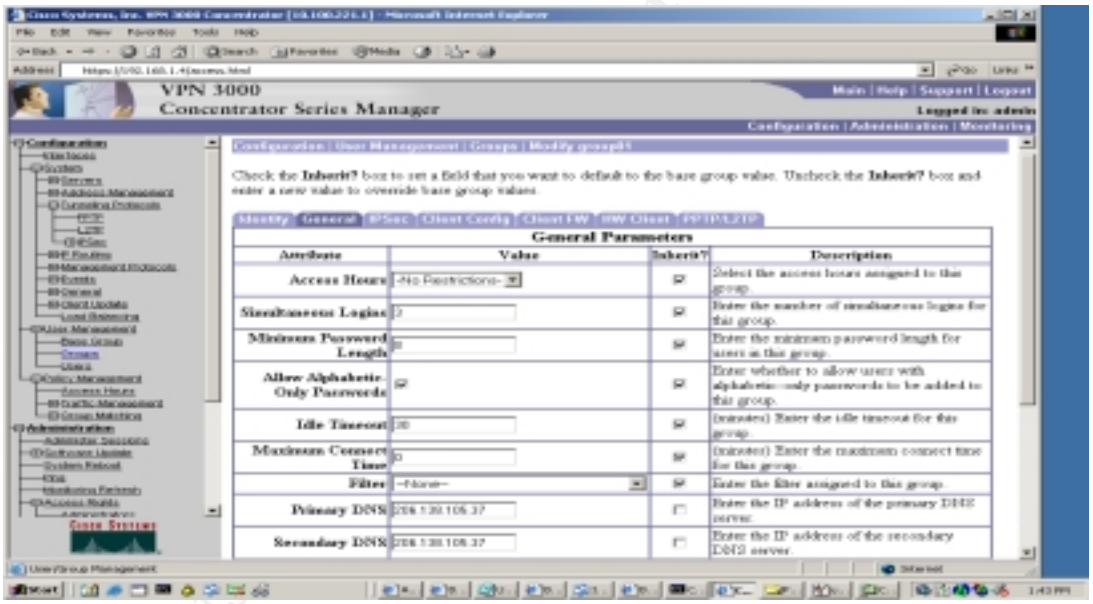


Figure 11 –Edition of the group

In the same property General we have chosen the **"IPSec"** only in **"Tunneling Protocol".**

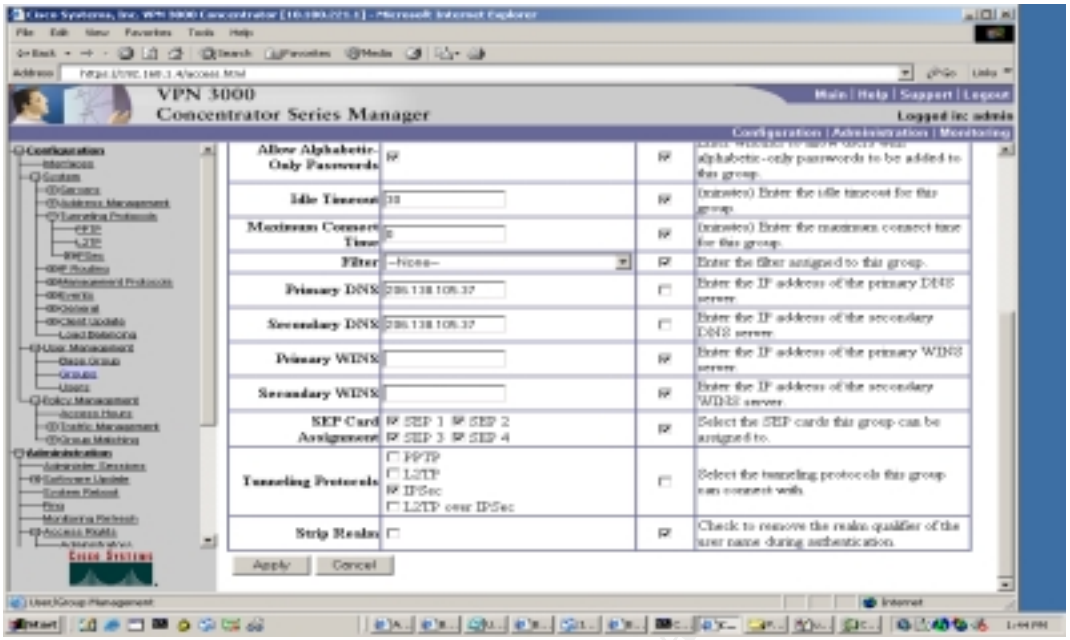


Figure 12 – Edition of the group – General

In the group in IP Sec Properties , we have chosen in IP Sec Security Associations “ESP-3DES-MD5”, which was modified to use to the digital certificate of the vpn server.

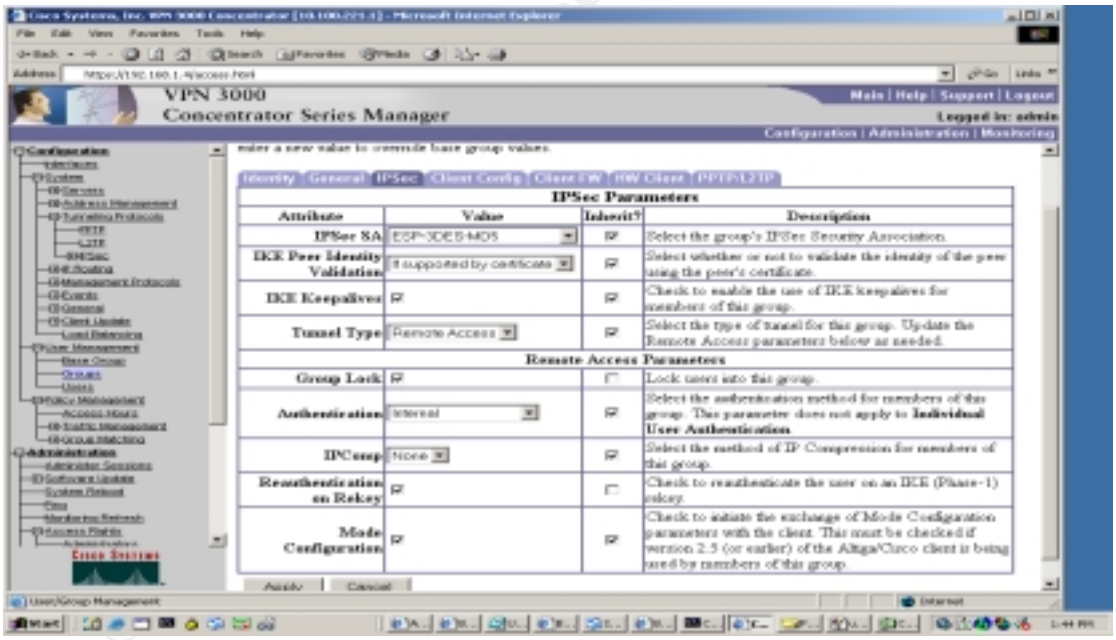


Figure 13- Edition of the group – IPsec

Then we have created a new user call **“supplier01”** but it´s user belong to the group **“group01”**, we assigned a static ip address to restrict the access or deny specific services.

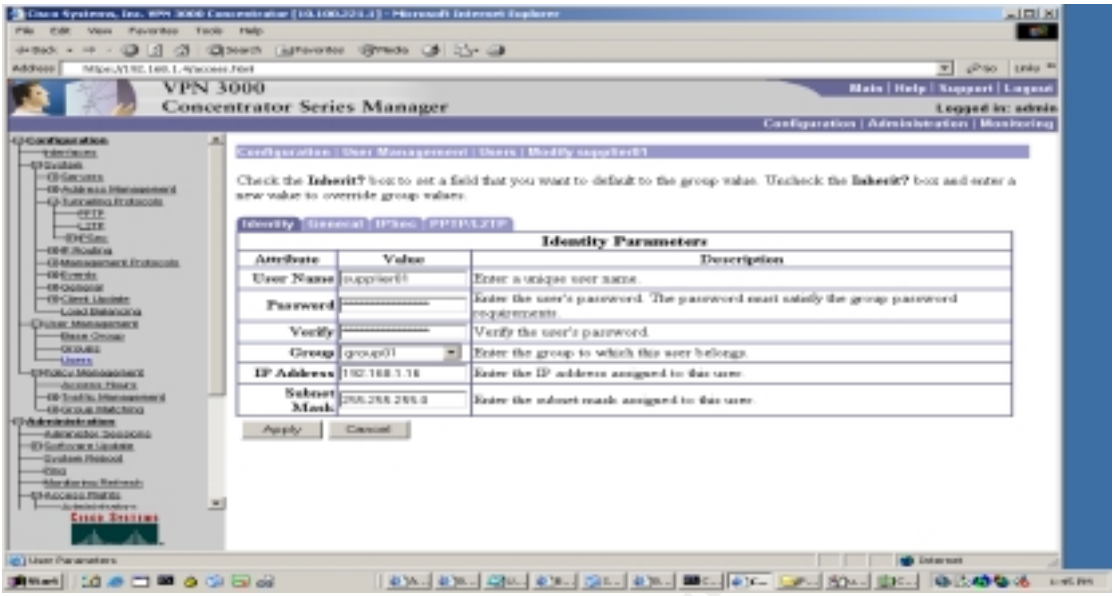


Figure 14 –Creation of the User

Step 6 : We have generated a digital certificate to the user **"supplier01"** in the CA server.

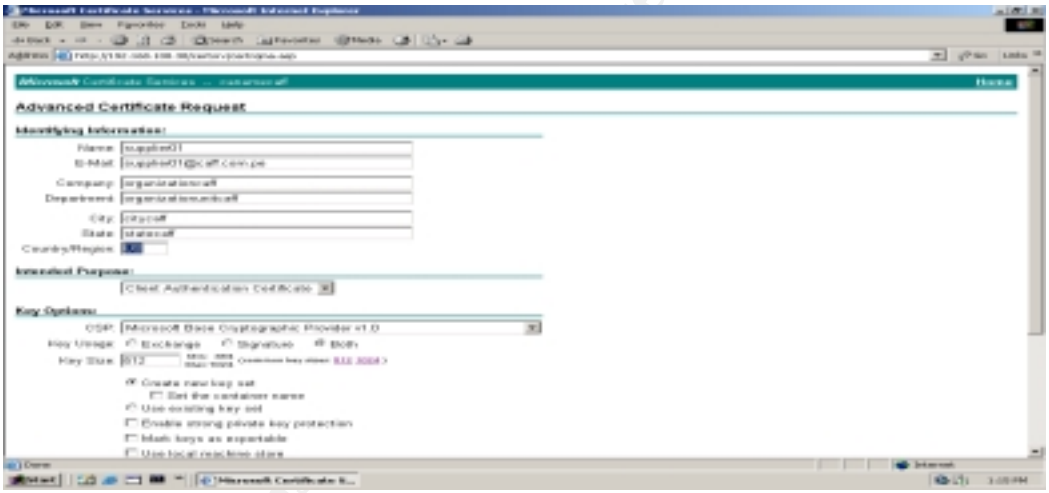


Figure 15 – Generation of the User ' Digital Certificate

Here we have showed the digital certificate to the user **"salespeople"**

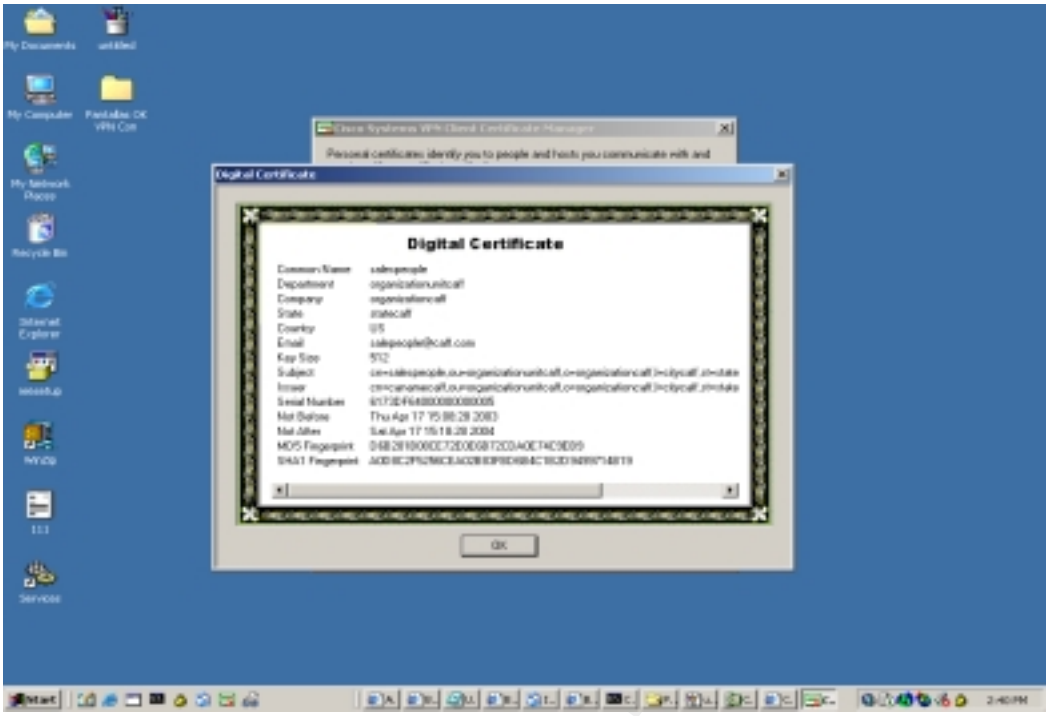


Figure 16 - Digital Certificate of the User

Step 7 : The configuration of vpn client in properties is with the respective group**"group01"** .And we connected to our vpn server and show the following:

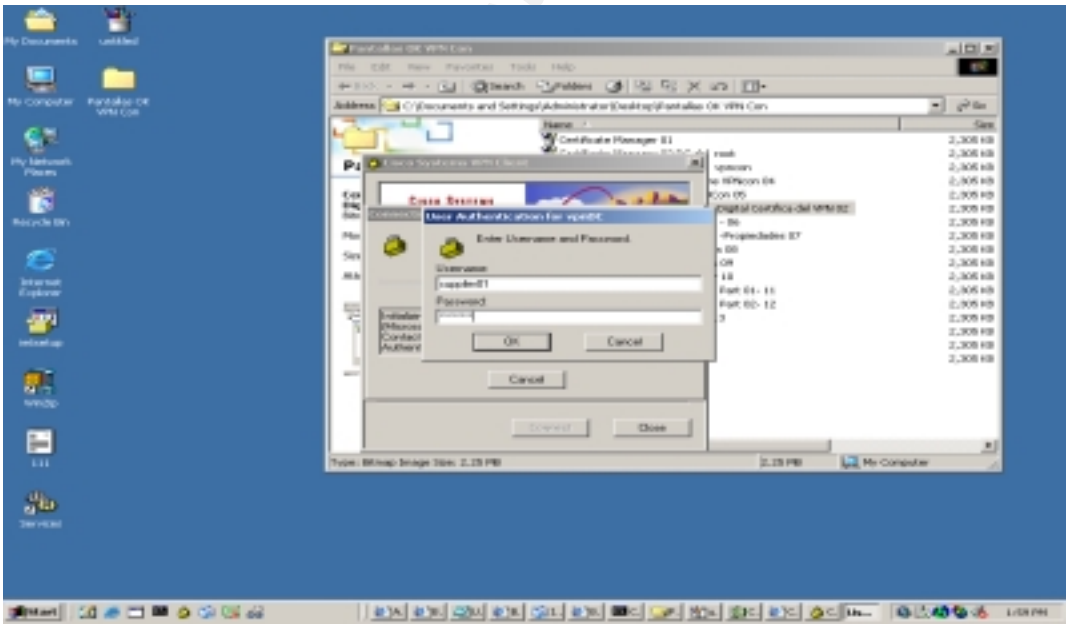


Figure 17 – User Authentication

Then we received the ip address "192.168.1.16" , range of the suppliers and partners at encryption 168 bits – 3DES.

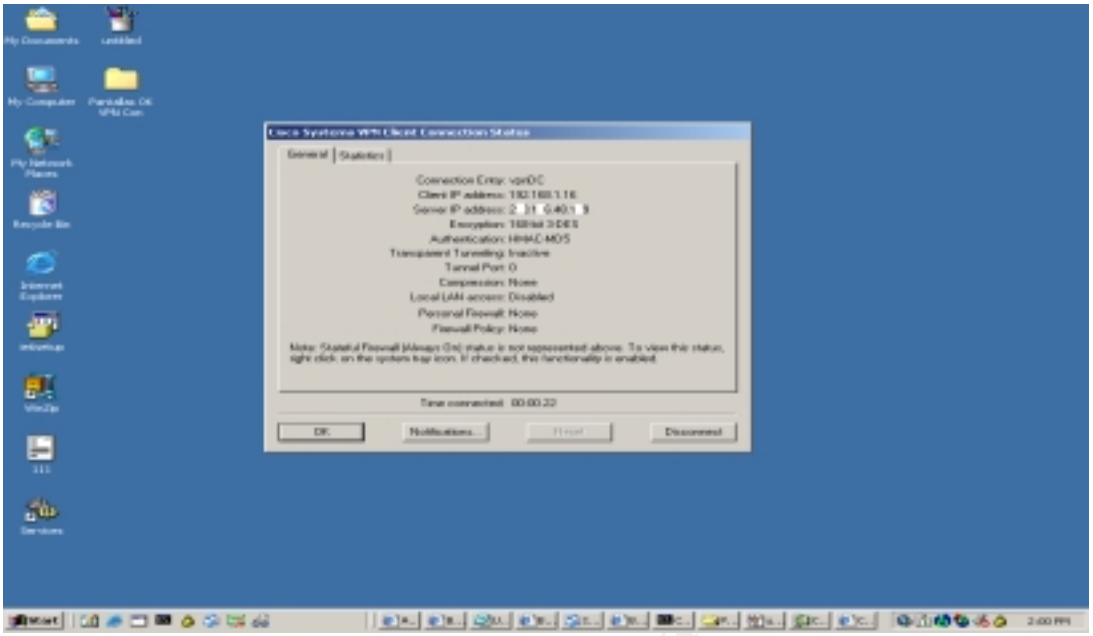


Figure 18 –Connections Status

In the following snapshot we are showing the "Live Event Log" that can see the process of authentication.

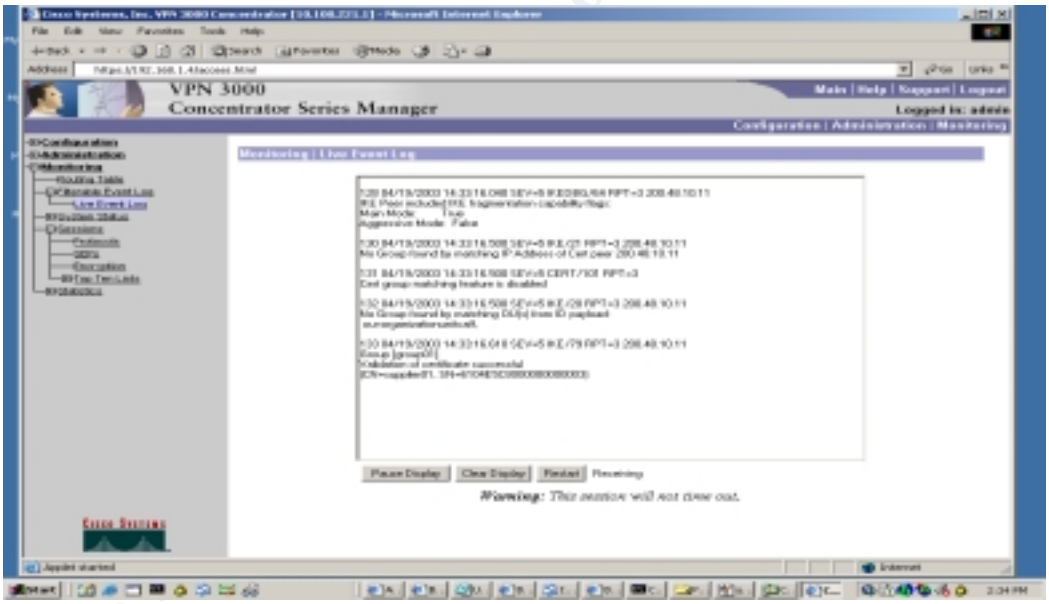


Figure 19 – Live Event Log

Step 8 : Finally we have configured to vpn server to that can send his logs to the Syslog server in the facility local 2.

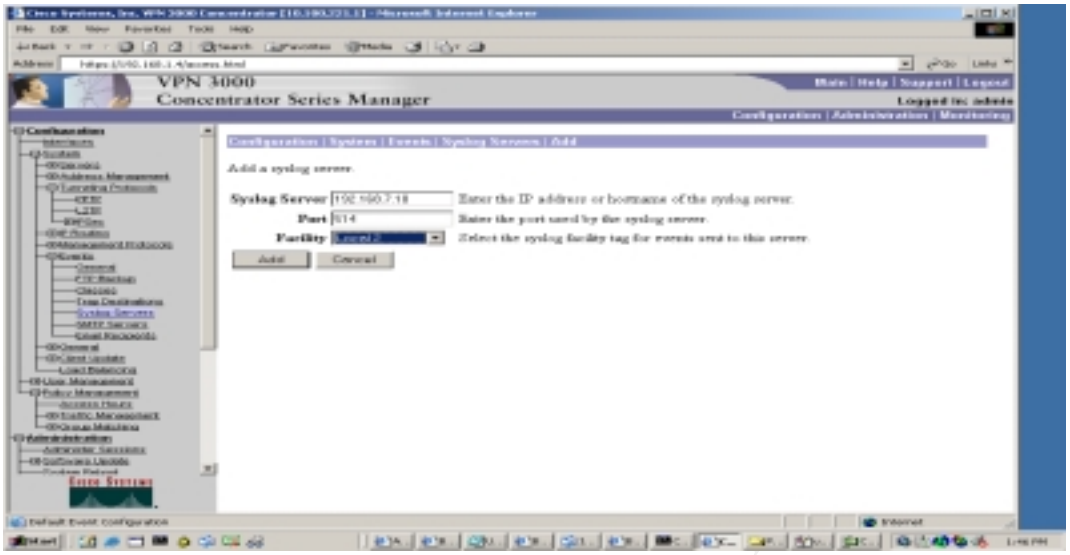


Figure 20 – Syslog Server

3.4. Tutorial for PIX Firewall and VPN Concentrator

3.4.1. PIX Firewall

CISCO PIX Firewall setup is not difficult , if the network topology is clear to the administrators. We are based it is topic about the following URL could be found at :

http://www.cisco.com/univercd/cc/td/doc/product/iaabu/pix/pix_v50/config/cfgforms.htm#41850).

Those forms are particularly designed for PIX firewall version 5.0; they are also

applicable to other versions. This tutorial will focus on security related topics. For detailed instruction on how to setup and management PIX 6.2, please refer to

http://www.cisco.com/univercd/cc/td/doc/product/iaabu/pix/pix_sw/v_62/index.htm

3.4.2. Steps :

3.4.2.1. Connect to PIX

As any other Cisco device, first of all, a terminal connection to PIX is needed for initial setup. In the case of hyper terminal, the configuration for serial port is: 9600 bits per second, 8 data bits, no parity, 1 stop bits and hardware flow control.

3.4.2.2. Configure PIX Firewall interfaces

CISCO PIX Firewall has four physical network interfaces.

Use :

ip address command to assign an ip address and netmask.

Interface *hardware_id hardware_speed* to identify the interface type.

nameif command to assign security levels on the interfaces.

pix# ***config t***

pix(config)# ***nameif ethernet0 outside security0***

pix(config)# ***nameif ethernet1 inside security100***

pix(config)# ***nameif ethernet2 dmz_ser security5***

pix(config)# ***nameif ethernet5 sa security55***

pix(config)# ***interface ethernet0 100full***

pix(config)# ***interface ethernet1 100full***

pix(config)# ***interface ethernet2 100full***

pix(config)# ***interface ethernet5 100full***

pix(config)# ***ip address inside 192.168.4.1 255.255.255.0***

pix(config)# ***ip address outside 200.20.x.2 255.255.255.240***

pix(config)# ***ip address dmz_ser 192.168.3.1 255.255.255.0***

pix(config)# ***ip address sfa 11.1.1.1 255.255.255.0***

For interfaces with a higher security level such as the inside interface, or a DMZ interface relative, use the **nat** and **global** commands to let users on the higher security interface access a lower security interface. For the opposite direction, from lower to higher, you use the **static** and **access-list** command. We will discuss this in great details later.

3.4.2.3. Failover

The failover ip address command statement for each interface to specify the standby unit's interface addresses. It is not necessary for the two units to be configured for this command to work correctly. The IP addresses on the standby unit are different from the active unit's addresses, but should be in the same subnet for each interface. The following example sets the IP addresses for the interfaces on the standby unit.

```
failover ip address inside 10.1.1.2
failover ip address outside 192.168.1.2
failover ip address intf2 192.168.2.2
failover ip address intf3 192.168.3.2
failover ip address 4th 172.16.1.2
```

Sample output from the show failover command shows that the secondary unit now has IP addresses for each interface:

```
show failover
Failover On
Cable status: Other side powered off
Reconnect timeout 0:00:00
Poll frequency 15 seconds
This host: primary - Active
Active time: 510 (sec)
Interface 4th (172.16.1.1): Normal (Waiting)
Interface intf3 (192.168.3.1): Normal (Waiting)
Interface intf2 (192.168.2.1): Normal (Waiting)
Interface outside (192.168.1.1): Normal (Waiting)
Interface inside (10.1.1.1): Normal (Waiting)
Other host: secondary - Standby
Active time: 0 (sec)
Interface 4th (172.16.1.2): Unknown (Waiting)
Interface intf3 (192.168.3.2): Unknown (Waiting)
Interface intf2 (192.168.2.2): Unknown (Waiting)
Interface outside (192.168.1.2): Unknown (Waiting)
Interface inside (10.1.1.2): Unknown (Waiting)
```

3.4.2.4. Stateful Failover

To configuring Stateful Failover, use the failover link command to specify the name of the dedicated interface you are using. For example, the “sfa” interface will be used for Stateful Failover and enter the following command.

failover link sfa

3.4.2.5. Routing

The following command sends any packets destined for the default route, to the router 200.20.x.1. In addition, add static routes for the networks that connect to the inside router as follows:

route outside 0.0.0.0 0.0.0.0 200.20.x.1 1

route inside 192.168.5.0 255.255.255.0 192.168.4.4

route inside 192.168.6.0 255.255.255.0 192.168.4.5

3.4.2.6. Static and Access List

By default, PIX deny any connectivity from lower security level interface to higher level interface. GIAC has to let outside user to access its server and servers in DMZ zone has to access internal servers.

Any server on a network that has a higher security level than the current interface requires a **static** and **access-list** command statement.

Static address translation creates a permanent, one-to-one mapping between a host on a higher security level interface and a global address on a lower security level interface. Static address translation hides the actual address of the server from users on the less secure interface, making casual access by unauthorized users less likely.

Syntax Description static

static [(*internal_if_name*, *external_if_name*)] *global_ip* *local_ip* [**netmask** *network_mask*]	
internal_if_name	The internal network interface name. The higher security level interface you are accessing.
external_if_name	The external network interface name. The lower security level interface you are accessing.
Global_ip	A global IP address. This address cannot be a PAT (Port Address Translation) IP address. The IP address on the lower security level interface you are accessing.
Netmask	Reserve word required before specifying the network mask.
network_mask	The network mask pertains to both *global_ip* and *local_ip*

Real Configuration static :

pix(config)#**static(dmz_ser, outside) 200.20.x.5 192.168.3.4 netmask 255.255.255.255 0 0**
pix(config)#**static(dmz_ser, outside) 200.20.x6 192.168.3.5 netmask 255.255.255.255 0 0**
pix(config)#**static(dmz_ser, outside) 200.20.x.7 192.168.3.6 netmask 255.255.255.255 0 0**
pix(config)#**static(dmz_ser, outside) 200.20.x8 192.168.3.7 netmask 255.255.255.255 0 0**

Syntax Description access-list

acl_ID	Name of an access list. You can use either a name or number
permit/deny	When used with the access-group command, the permit option selects a packet to traverse the PIX Firewall. While the deny option does not allow a packet to traverse the PIX Firewall. By default, PIX Firewall denies all inbound or outbound packets unless you specifically permit access.
Port	Services you permit or deny access to.

Protocol	Name or number of an IP protocol. It can be one of the keywords icmp, ip, tcp, or udp, or an integer in the range 1 to 254 representing an IP protocol number. To match any Internet protocol, including ICMP, TCP, and UDP, use the keyword ip.
source_addr	Address of the network or host from which the packet is being sent.
source_mask	Netmask bits (mask) to be applied to *source_addr*, if the source address is for a network mask.
Remote_addr	IP address of the network or host remote to the PIX Firewall.
Remote_mask	Netmask bits (mask) to be applied to *remote_addr*, if the remote address is a network mask.

Real Configuration access-list :

```
pix(config)#access-list 101 permit tcp any  200.20.x.5 eq 80
pix(config)#access-list 101 permit tcp any  200.20.x.5 eq 8080
pix(config)#access-list 101 permit tcp any  200.20.x.6 eq 80
pix(config)#access-list 101 permit tcp any  200.20.x.6 eq 8080
pix(config)#access-list 101 permit tcp any  200.20.x.7 eq 25
pix(config)#access-list 101 permit udp any  200.20.x.8 eq 53
pix(config)#access-list 101 permit tcp any  200.20.x.8 eq 53
#Apply the access-list 101 with the access-group command in the
#interface outside
pix(config)#access-group 101 in interface outside
```

3.4.3. VPN Concentrator

3.4.4. Steps :

3.4.4.1. Connect to VPN Concentrator

As any other Cisco device, first of all, a terminal connection to VPN Concentrator is needed for initial setup. In the case of hyper terminal, the configuration for serial port is: 9600 bits per second, 8 data bits, no parity, 1 stop bits and hardware flow control.

3.4.4.2. Initial Configuration

You should then see the following

```
Login:
```

Type in the default username and password, which are both admin and then hit enter. You should then see the following:

```
Quick -> [ 17:26:14 ] _
```

Type the correct time in 24-hour format and then hit enter. The next screen should ask you to enter the correct date in MM/DD/YYYY format:

```
-- : Enter the date ...

> Date
Quick -> [ 03/26/2001 ] _
```

Once the correct date has been input hit enter again. The time zone needs to be set. In the case of GIAC the time zone should be set to –5. Enter –5 and then hit enter.

```
-- : Set the time zone on your device. ...
-- : Enter the time zone using the hour offset from GMT: ...
> Time Zone
Quick -> [ 0 ] _
```

Daylight Savings Time Support should be disabled so input 2 and hit enter.

```
1) Enable Daylight Savings Time Support
2) Disable Daylight Savings Time Support
Quick -> [ 2 ] _
```

The next screen that will appear will ask for the IP address for interface 1. Interfaces on the VPN are numbered starting from 1 (as opposed to 0).

```
This table shows current IP addresses.
Interface IP Address/Subnet Mask MAC Address
-----------------------------------------------------------------
```

```
| Ethernet 1 - Private | 0.0.0.0/0.0.0.0 |
| Ethernet 2 - Public | 0.0.0.0/0.0.0.0 |
| Ethernet 3 - External | 0.0.0.0/0.0.0.0 |
-----------------------------------------------------------------
** An address is required for the private interface. **
> Enter IP Address
Quick Ethernet 1 -> [ 0.0.0.0 ] _
```

Type in the following IP Address for Ethernet 1, the private interface,192.168.1.4 then hit enter. The concentrator will prompt for the subnet mask. Input 255.255.255.0 and then hit enter again.

```
> Enter Subnet Mask
Quick Ethernet 1 -> [ 255.0.0.0 ] _
```

The concentrator will prompt for the line speed. Select 2 and hit enter.

```
1) Ethernet Speed 10 Mbps
2) Ethernet Speed 100 Mbps
3) Ethernet Speed 10/100 Mbps Auto Detect
Quick -> [ 2 ] _
```

Select full duplex on the next screen by typing 2 and hitting enter.

```
1) Enter Duplex - Half/Full/Auto
2) Enter Duplex - Full Duplex
3) Enter Duplex - Half Duplex
Quick -> [ 2 ] _
```

The next screen should look like this:

```
1) Modify Ethernet 1 IP Address (Private)
2) Modify Ethernet 2 IP Address (Public)
3) Modify Ethernet 3 IP Address (External)
4) Configure Expansion Cards
5) Save changes to Config file
6) Continue
7) Exit
Quick -> _
```

Type 5 and hit enter to save changes. Then type 7 to exit. Now place the crossover cable into the private interface and hook it into the laptop and type in your web browser http://192.168.1.4/access.html, then type the default login and password into the appropriate boxes both of them are admin. Then select the hyperlink labeled **"Click Here to Start Quick Configuration"** to configured the Public Interface, Click the Ethernet 2 interface. The **"Configuration>Quick>IP Interfaces>Ethernet 2"**, then Click the **"Static IP"**, Fill in the following parameters to configure Ethernet 2.

IP Address: 200.20.X.4
Subnet Mask: :255.255.255.240
Filter : None
Speed: 100
Duplex :Full

Click continue. The next screen that appears will be the "**Configuration|Quick|system info Screen"**. On this screen the following parameters need to be loaded.

DNS Server : 200.x.x.x
Domain : giac.com.pe
Default Gateway : 200.20.X.1

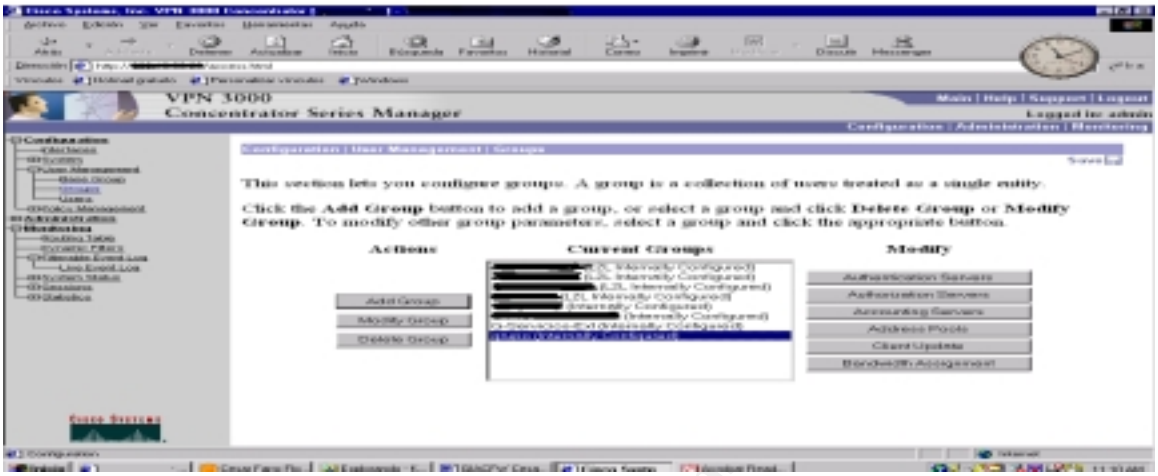
Click continue. The **Configuration > Quick > Protocols"** screen will appear. On this screen only leave IPSec checked then click continue. The next screen will be the **"Configuration > Quick > Address Assignment"** screen. On this screen the Configured Pool checkbox should be selected. Click Continue.

The **"Configuration > Quick > Authentication"** Screen will appear next. Internal Server is the authentication set by default.

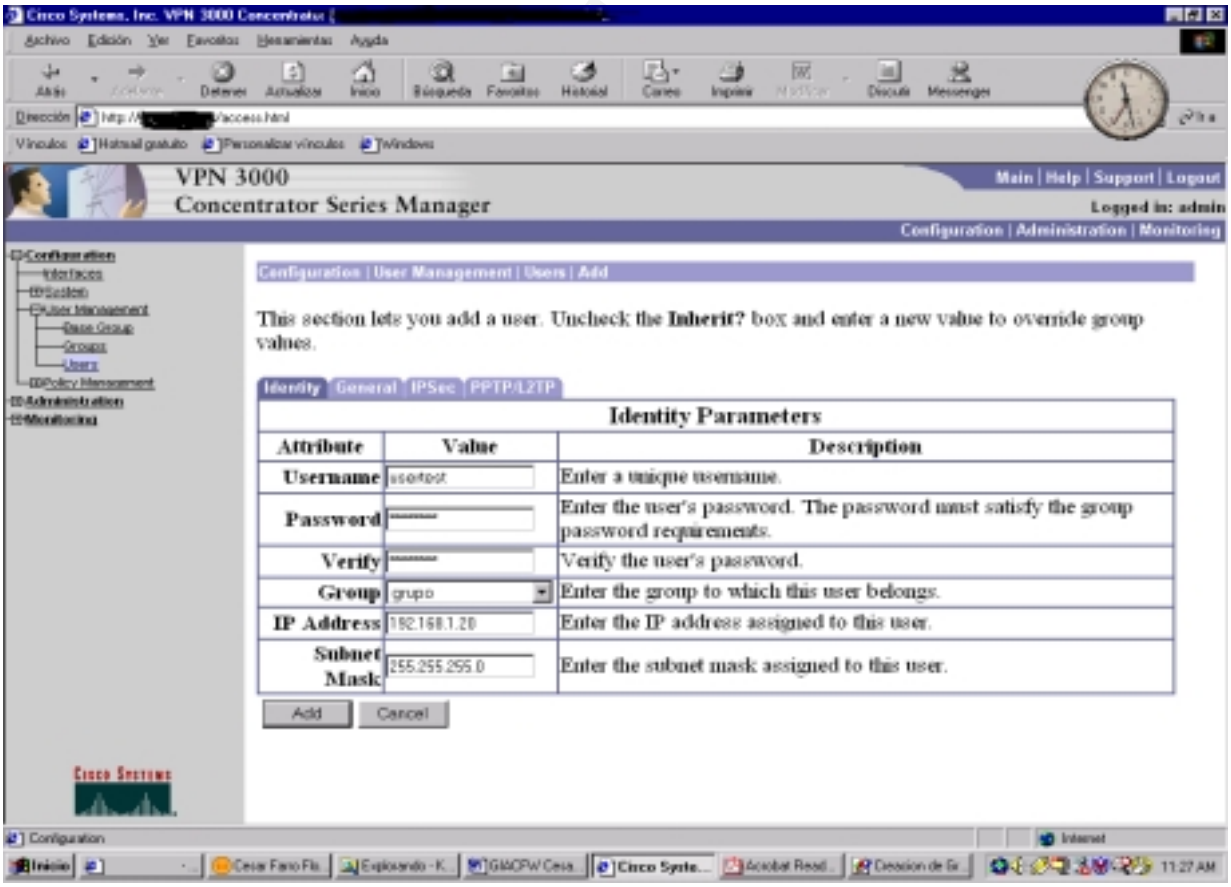
The next screen is **"Configuration > Quick > IPSec Group"**. This screen sets the IPSec Group. The Group Name is **"testgroup"**. The password should again conform to the password policy and again has to be verified. Click continue. **"Configuration > Quick > Admin Password"** allows for the password to be reset. This should be done at this point. The password should be 16 characters and consist of no dictionary crackable words. Verify the password then click continue.

The final screen in quick configuration is **"Configuration > Quick > Done"**. Click **"Save Needed"** icon at the upper right corner of the window to save the active configuration. A web browser window should appear with the words **"Save Successful"**. Click OK in that window to dismiss it and then click the hyperlinked labeled "Configuration" on the **"Configuration > Quick > Done"** screen.

Then You can created a new group and users **"Configuration > User Managements > Groups and Users"**



Creation Groups



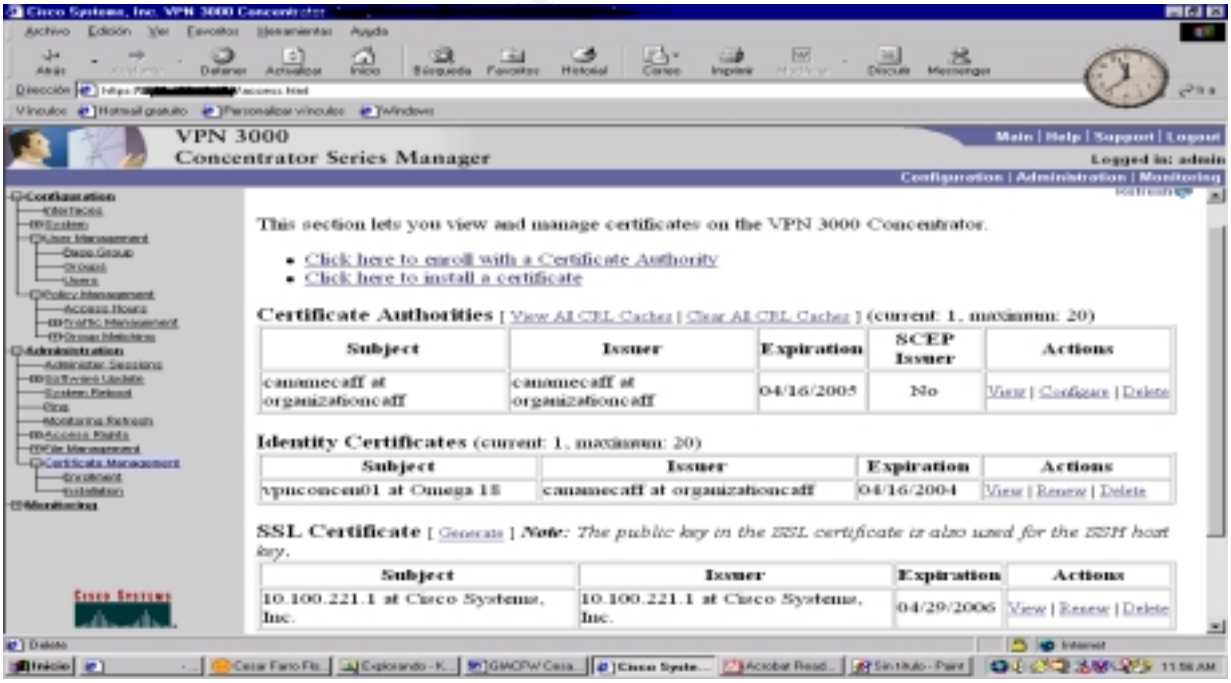
Creation Users

3.4.4.3. Installation Digital Certificate

To the installation of the digital certificate you have to fill the following dates:

Common Name (CN)	Altiga30
Organization Unit (OU)	group **"The name of the Organization Unit must be the name of the group that you created "**
Organization (O)	GIAC Enterprise
Locality(L)	Lima
State/Province(SP)	Lima
Country	PE
Alternative Name (FQDN)	vpn.giac.com.pe
Key size	RSA 512 bits

And generated a **"pkcs.txt"**. With Certificate Authority Server you can generated the digital certificate to the vpn a due the pkcs.txt. Finally you must install the digital certificate in the VPN Concentrator. [4]



Installation Digital Certificate

[4] http://www.cisco.com/en/US/products/hw/vpndevc/ps2284/products_configuration_example09186a00800946f1.shtml

3.5. Security Server

Because it is very important to have our severs always available to our customers, partners and suppliers, giving all the security levels that they need, in this section we will describe a group of "good practices" which help us in the process of installation of our servers. Remember that the availability of our servers has a direct result in our business.

This section shows our "internal procedures" in order to install our servers, giving suggestions for several topics like: the internal firewall, the web server, the dns server, the mail server, etc..

First of all, we will begin with the process of installation of the operating system. After that, we will cover the process of installation and configuration of the gateway firewall (or an standalone one) over Linux.

3.5.1. Procedure for the installation of the Operating System

3.5.1.1. Operating System

We will use Red Hat Linux version 8.0, which is latest until March 24th of 2003, for the following servers: web, mail, dns and for internal firewalls.

We have chosen this distribution of Linux because it is widely used and it is well known for system and network administrators. Also, it has a good errata support through their web page and Red Hat Network.

3.5.1.2. Installation

The installation only must consider the necessary packages for the bastion host, for this reason it is strongly recommended to choose "Custom installation":

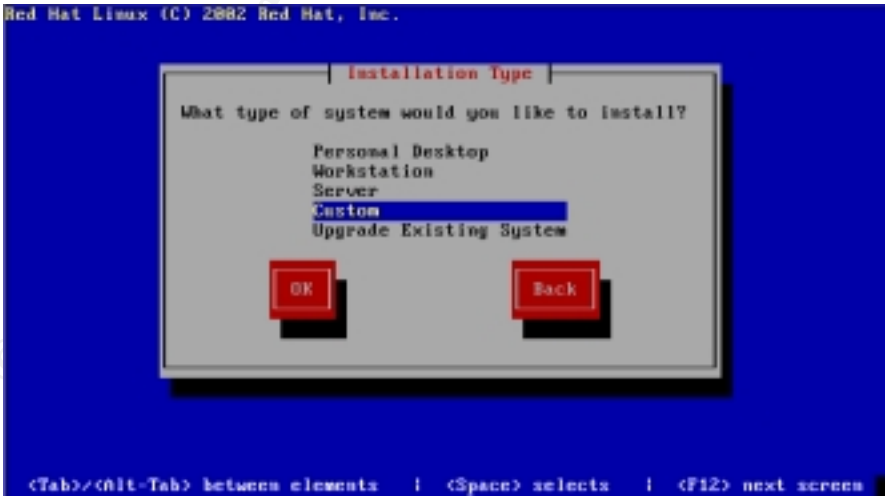


Figure 1.1 - Red Hat 8.0 Installation Type

Also, we need boot loader security. This is done by selecting GRUB, where we must set a password of 14 characters of length:

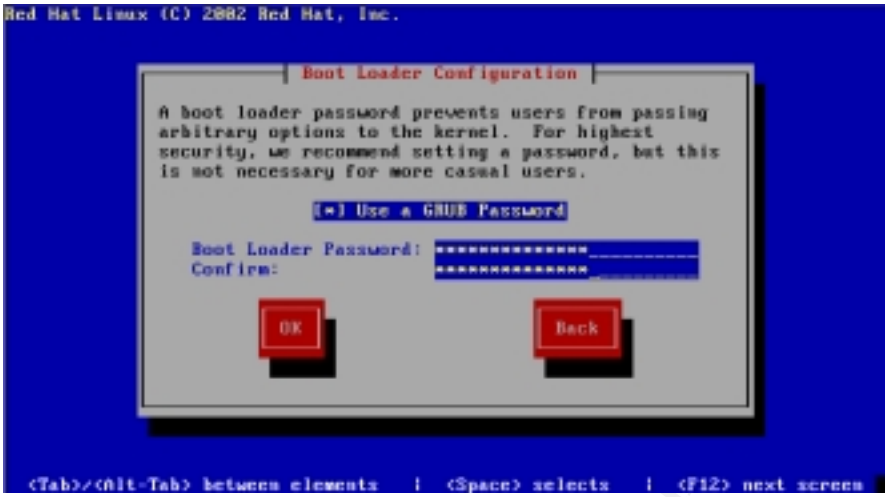


Fig. 1.2 - GRUB Password Protection

In the firewall configuration screen, select “No firewall” because instead of the default firewall template we will use a custom standalone script.

Fig 1.3 - Firewall Configuration

For passwords management is recommended to use this options:

- Use Shadow passwords.
- Enable MD5 passwords.

In the package group selection we must not select any package group. Note that the overall size is 476 Mb, this is still higher for a firewall configuration but we will remove some packages after the installation.

Fig 1.4 - Package Group Selection

All other configurations are dependent of system requirements and hardware configuration (not are related to security issues). So, follow according to your own requirements.

3.5.1.3. Disable Services

There are unnecessary services enabled by default, we have to disable them and leave only the minimum required for our configuration.

Using the tool "ntsysv" we will configure the host with the necessary services.

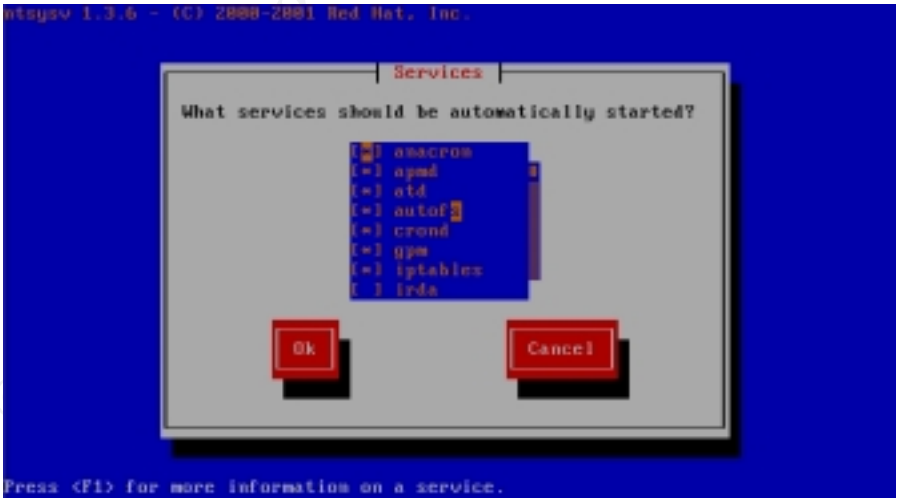


Fig 1.5 - Services Configuration with Ntsysv

The required services and their description are shown bellow:

Service	General description
Cron	It is an standard UNIX program that runs user-specified programs at periodic scheduled times.
Keytable	Loads the selected keyboard map as set it /etc/keyboard.
Kuduzu	Runs hardware probe, and optionally configures changed hardware.
Network	Configures the network interfaces at boot time.
andom	Saves and restore system entropy pool for higher quality random number generation.
Sshd	We will use Secure Shell for remote access, leave this option selected.

Is strongly recommended to inactivate: portmap, nfs and sendmail. Those programs are configured to be loaded by default.

Before disable the unnecessary services the output of the command netstat –an –A inet, must shows something like this:

```
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:1024            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:111             0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:25            0.0.0.0:*               LISTEN
udp        0      0 0.0.0.0:1024            0.0.0.0:*
udp        0      0 0.0.0.0:111             0.0.0.0:*
```

After having disabled the unnecessary services and reboot the output of the command netstat –an –A inet, shows:

```
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
```

Before disable the unnecessary services the output of the command ps -aux, shows:

```
USER       PID %CPU %MEM   VSZ  RSS TTY      STAT START   TIME COMMAND
root         1  4.6  0.3  1336  480 ?        S    05:13   0:04 init
root         2  0.0  0.0     0    0 ?        SW   05:13   0:00 [keventd]
root         3  0.0  0.0     0    0 ?        SW   05:13   0:00 [kapmd]
root         4  0.0  0.0     0    0 ?        SWN  05:13   0:00 [ksoftirqd_CPU0]
root         5  0.0  0.0     0    0 ?        SW   05:13   0:00 [kswapd]
root         6  0.0  0.0     0    0 ?        SW   05:13   0:00 [bdflush]
root         7  0.0  0.0     0    0 ?        SW   05:13   0:00 [kupdated]
root         8  0.0  0.0     0    0 ?        SW   05:13   0:00 [mdrecoveryd]
root        16  0.4  0.0     0    0 ?        SW   05:13   0:00 [kjournald]
root        75  0.0  0.0     0    0 ?        SW   05:13   0:00 [khubd]
root       439  0.3  0.4  1400  536 ?        S    05:14   0:00 syslogd -m 0
root       443  0.0  0.3  1336  428 ?        S    05:14   0:00 klogd -x
rpc        460  0.0  0.4  1484  532 ?        S    05:14   0:00 portmap
rpcuser    479  0.0  0.5  1528  724 ?        S    05:14   0:00 rpc.statd
root       544  0.0  0.3  1328  476 ?        S    05:14   0:00 /usr/sbin/apmd -
p
root       582  0.3  1.1  3276 1468 ?        S    05:14   0:00 /usr/sbin/sshd
root       602  0.0  1.7  5040 2264 ?        S    05:14   0:00 sendmail:
accepti
smmsp      612  0.0  1.6  4856 2048 ?        S    05:14   0:00 sendmail: Queue
r
root       622  0.0  0.3  1372  428 ?        S    05:14   0:00 gpm -t ps/2 -m
/d
root       631  0.0  0.4  1512  612 ?        S    05:14   0:00 crond
root       640  0.0  0.4  1360  552 ?        SN   05:14   0:00 anacron -s
daemon     649  0.0  0.4  1368  520 ?        S    05:14   0:00 /usr/sbin/atd
root       658  0.1  0.8  2264 1040 ?        S    05:14   0:00 login -- root
root       659  0.0  0.3  1316  404 tty2      S    05:14   0:00 /sbin/mingetty
tt
root       660  0.0  0.3  1316  404 tty3      S    05:14   0:00 /sbin/mingetty
tt
root       661  0.0  0.3  1316  404 tty4      S    05:14   0:00 /sbin/mingetty
tt
root       662  0.0  0.3  1316  404 tty5      S    05:14   0:00 /sbin/mingetty
tt
root       663  0.0  0.3  1316  404 tty6      S    05:14   0:00 /sbin/mingetty
tt
root       666  1.0  1.1  4144 1424 tty1      S    05:15   0:00 -bash
```

After having disabled the unnecessary services and reboot the output of the command ps -aux, shows:

```
USER       PID %CPU %MEM   VSZ  RSS TTY      STAT START   TIME COMMAND
root         1  6.0  0.3  1336  480 ?        S    05:16   0:04 init
root         2  0.0  0.0     0    0 ?        SW   05:16   0:00 [keventd]
root         3  0.0  0.0     0    0 ?        SW   05:16   0:00 [kapmd]
root         4  0.0  0.0     0    0 ?        SWN  05:16   0:00 [ksoftirqd_CPU0]
root         5  0.0  0.0     0    0 ?        SW   05:16   0:00 [kswapd]
root         6  0.0  0.0     0    0 ?        SW   05:16   0:00 [bdflush]
root         7  0.0  0.0     0    0 ?        SW   05:16   0:00 [kupdated]
root         8  0.0  0.0     0    0 ?        SW   05:16   0:00 [mdrecoveryd]
root        16  0.5  0.0     0    0 ?        SW   05:16   0:00 [kjournald]
root        75  0.0  0.0     0    0 ?        SW   05:16   0:00 [khubd]
root       422  0.3  0.4  1400  536 ?        S    05:17   0:00 syslogd -m 0
root       426  0.1  0.3  1336  428 ?        S    05:17   0:00 klogd -x
root       463  0.2  1.1  3276 1468 ?        S    05:17   0:00 /usr/sbin/sshd
root       474  0.0  0.4  1512  612 ?        S    05:17   0:00 crond
root       481  0.3  0.8  2264 1040 ?        S    05:17   0:00 login -- root
root       482  0.0  0.3  1316  404 tty2     S    05:17   0:00 /sbin/mingetty
tt
root       483  0.0  0.3  1316  404 tty3     S    05:17   0:00 /sbin/mingetty
tt
root       484  0.0  0.3  1316  404 tty4     S    05:17   0:00 /sbin/mingetty
tt
root       485  0.0  0.3  1316  404 tty5     S    05:17   0:00 /sbin/mingetty
tt
root       486  0.0  0.3  1316  404 tty6     S    05:17   0:00 /sbin/mingetty
tt
root       489  1.1  1.1  4144 1424 tty1     S    05:17   0:00 –bash
```

3.5.1.4. Removing Packages

The install program has installed 233 packages, some of them are unnecessary for the bastion host configuration. The following script removes those unnecessary packages, using RPM.

```
#!/bin/sh

rpm -e acl
rpm -e anacron
rpm -e apmd
rpm -e at
rpm -e attr
rpm -e autofs
rpm -e bind-utils
rpm -e dhclient
rpm -e dos2unix
```

```
rpm -e finger
rpm -e ftp
rpm -e lftp
rpm -e lokkit
rpm -e mailcap
rpm -e minicom
rpm -e mouseconfig
rpm -e net-snmp-utils
rpm -e net-snmp
rpm -e parted
rpm -e --nodeps procmail
rpm -e sendmail
rpm -e rdate
rpm -e rdist
rpm -e rsync
rpm -e stunnel
rpm -e talk
rpm -e tcpdump
rpm -e tcp_wrappers
rpm -e telnet
rpm -e unix2dos
rpm -e wget
rpm -e whois
rpm -e wireless-tools
rpm -e zip
rpm -e gnome-libs
rpm -e esound
rpm -e rp-pppoe
rpm -e wvdial
rpm -e ppp
rpm -e imlib
rpm -e gtk+
rpm -e libungif
rpm -e --nodeps XFree86-Mesa-libGL
rpm -e XFree86-libs
rpm -e --nodeps yp-tools
rpm -e ypbind
rpm -e ORBit
rpm -e aspell
rpm -e audiofile
rpm -e make
rpm -e netconfig
rpm -e hesiod
rpm -e isdn4k-utils
rpm -e lilo
```

```
rpm -e mtr
```

3.5.2. *Updating the System*

Refers to Red Hat Errata Support.

3.5.3. *Secure Shell Configuration*

We are using the program ssh 3.1p1, and in the file **"ssd_config"** we have configured the followings lines:

/etc/ssh/ssd_config

Protocol 2

By default SSH (the program) allows both versions 1 and 2 of the SSH protocol. #Uncomment the line **"#Protocol 2,1"**, by removing the leading "#" and change it #to say "Protocol 2" this disables Protocol version 1 of SSH

Password Authentication no

By default OpenSSH allows for two methods of authentication, one is a key based method where the users stores their public key on the server and logs in authenticating themselves using their private key, the other is **"password authentication",** where the user simply provides their username and password. While the password authentication can be more convenient (It is more portable since a user does not have to hold their private key with them) it is less secure since it only requires knowledge of the username and password. On the contrary, the key method requires the possession the private key and the knowing of the passphrase which protects the private key.

OpenSSH comes with a method of tunneling Xwindows windows through the SSH protocol.This should be disabled (find the line **"X11forwarding yes"** and **delete the line)**.

3.6. Third line of Defense: Internal Firewalls

3.6.1. *Internal Database Firewall : Iptables Firewall Configuration*

The following diagram shows the topology used in order to protect the database server. There is an internal firewall which runs over Linux 8.0, and will be in charge of separate the data base network from the firewalls network.

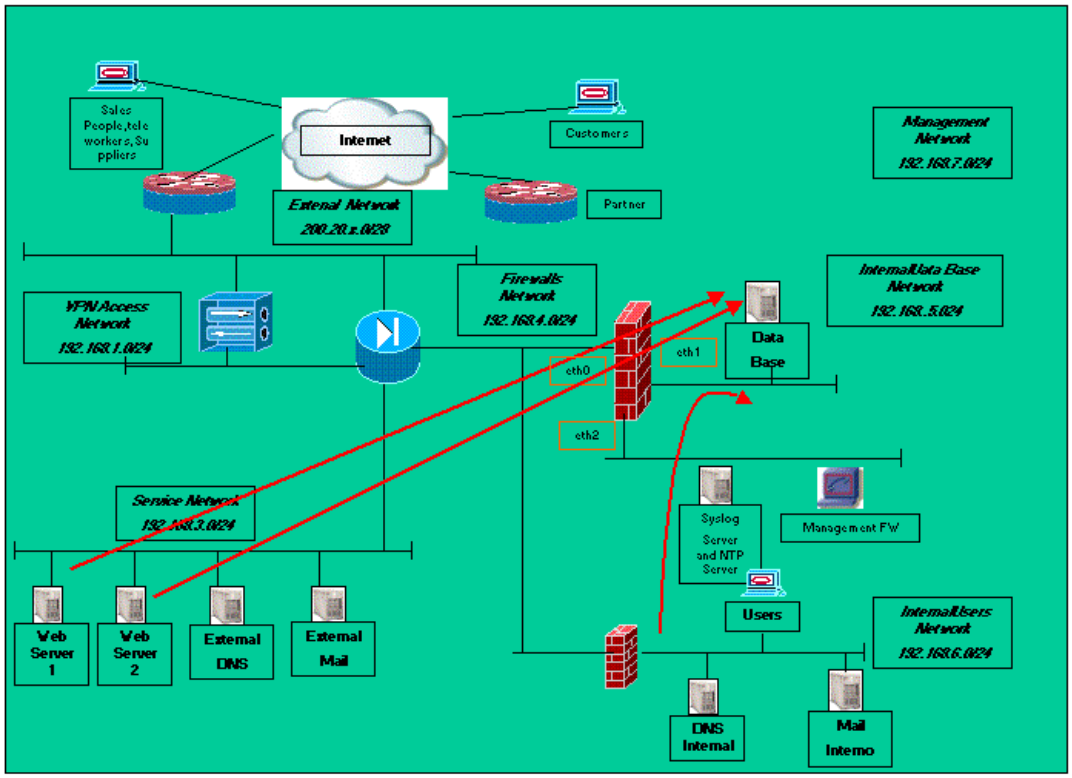


Figure 1 – Data Base Firewall

Principal commands used during the configuration
 # Related Connections
`iptables -A FORWARD –m state –state ESTABLISHED, RELATED –j ACCEPT`
 # Connection to Data Base from Web Server 1
iptables –A FORWARD –i eth0 –o eth1 -s 192.168.3.4 –d 192.168.5.2 –p tcp - -sport 1024: - -dport 1521 –m state - - state NEW -j ACCEPT
 # Connection to Data Base from Web Server 2
iptables –A FORWARD –i eth0 –o eth1 -s 192.168.3.5 –d 192.168.5.2 –p tcp - -sport 1024: - -dport 1521 –m state - - state NEW -j ACCEPT
 # Connection to Data Base from the Developers
`iptables –A FORWARD –i eth1 –o eth0 -s 192.168.6.10 –d 192.168.5.2 –p tcp - -sport 1024: - -dport 1521 –m state - - state NEW -j ACCEPT`
`iptables –A FORWARD –i eth1 –o eth0 -s 192.168.6.11 –d 192.168.5.2 –p tcp - -sport 1024: - -dport 1521 –m state - - state NEW -j ACCEPT`
`iptables –A FORWARD –i eth1 –o eth0 -s 192.168.6.12 –d 192.168.5.2 –p tcp - -sport 1024: - -dport 1521 –m state - - state NEW -j ACCEPT`

3.6.2. Internal Users Firewall : Iptables Firewall Configuration

The following topology shows an internal firewall which runs over Linux 8.0 and is in charge of separate the internal users network from the firewall network.

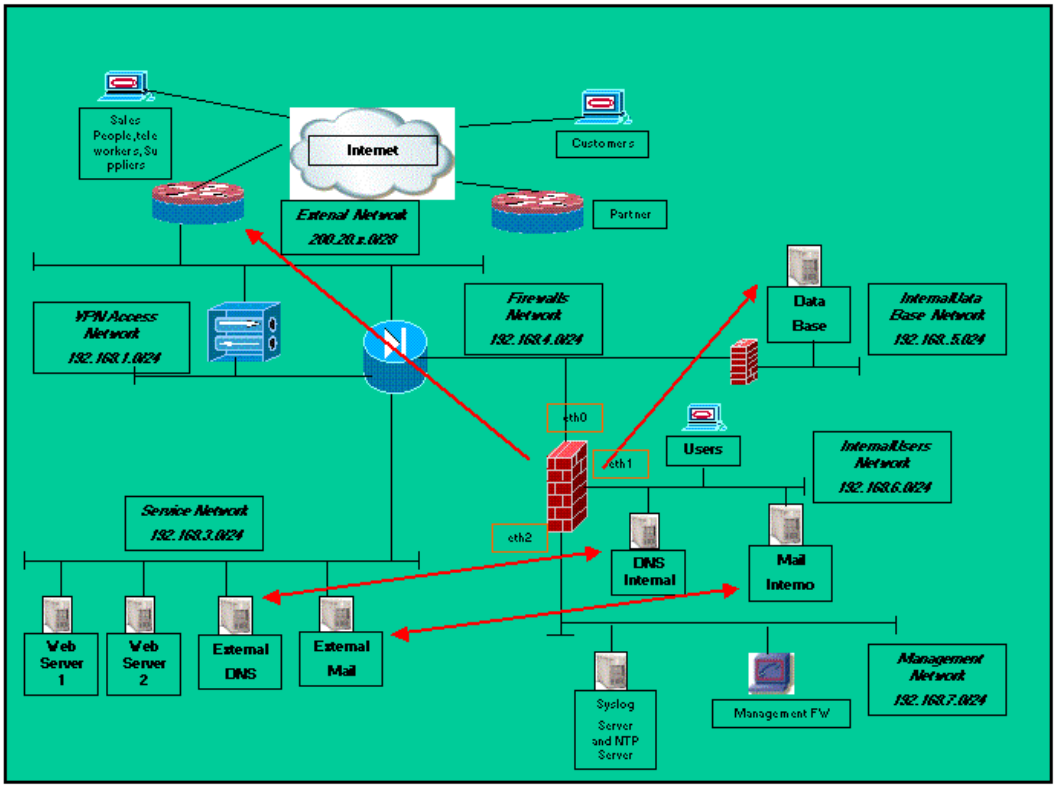


Figure 2 - Internal Users Firewall

Principal command used during the configuration

 # To Transparent Proxy Cache

```
iptables –t nat –A PREROUTING  -i eth1 –s 192.168.6.0/24 –p tcp - -
sport 1024: - -dport 80 –j REDIRECT - - to –ports 3128
```

 # Related Connections

```
iptables  -A  FORWARD –m state –state ESTABLISHED, RELATED –j ACCEPT
```

 # Traffic DNS between Internal DNS and External DNS.

```
iptables –A FORWARD –i eth1 –o eth0  -s 192.168.6.2 –d 192.168.3.6 –p
udp   - -sport  1024:  - -dport 53  –m state  - - state NEW  -j ACCEPT
```

 # Relay SMTP between Internal SMTP and External SMTP.

```
iptables –A FORWARD –i eth1 –o eth0  -s 192.168.6.3 –d 192.168.3.7 –p
tcp  - -sport  1024:  - -dport 25 –m state  - - state NEW  -j ACCEPT
```

3.6.3. *Model of script based in "Iptables Firewall"*

The following script is based in iptables firewall . This script will be used as a model in the process of configuration of the stand alone firewall and the gateway firewall.

```
#!/bin/sh
# set -x
IPT="/sbin/iptables"
# Clear the old rules
$IPT -F
$IPT -X
# Definition of Policies by Default
$IPT -P INPUT DROP
$IPT -P OUTPUT DROP
$IPT -P FORWARD DROP

#Permit interface LOOPBACK
$IPT -A INPUT -i lo -j ACCEPT
$IPT -A OUTPUT -o lo -j ACCEPT
# Chains created by User
$IPT -N LOG_DROP_PSCAN
$IPT   -A   LOG_DROP_PSCAN   -j   LOG   --log-level   info   --log-prefix
"PORT_SCAN"
$IPT -A LOG_DROP_PSCAN -j DROP
$IPT -N LOG_DROP_IPINVALIDO
$IPT -A LOG_DROP_IPINVALIDO -j LOG --log-level info --log-prefix
"IP_INVALIDO"
$IPT -A LOG_DROP_IPINVALIDO -j DROP


#
# INPUT Chain
# Blocked packets that have combinations of bits invalids TCP  used #
to
scan ports in mode stealth
#
# All the bits in 0
$IPT -A INPUT -p tcp --tcp-flags ALL NONE -j LOG_DROP_PSCAN
# SYN and FIN in 1
$IPT -A INPUT -p tcp --tcp-flags SYN,FIN SYN,FIN -j LOG_DROP_PSCAN
```

```
# SYN and RST in 1
$IPT -A INPUT -p tcp --tcp-flags SYN,RST SYN,RST -j LOG_DROP_PSCAN
# FIN and RST in 1
$IPT -A INPUT -p tcp --tcp-flags FIN,RST FIN,RST -j LOG_DROP_PSCAN
# FIN is the only bit established, without ACK
$IPT -A INPUT -p tcp --tcp-flags ACK,FIN FIN -j LOG_DROP_PSCAN
# PSH is the only bit established, without ACK
$IPT -A INPUT -p tcp --tcp-flags ACK,PSH PSH -j LOG_DROP_PSCAN
# URG is the only bit established, without ACK
$IPT -A INPUT -p tcp --tcp-flags ACK,URG URG -j LOG_DROP_PSCAN
# Accept connections Established or Relational
$IPT -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
# Blocked connections Invalids
$IPT -A INPUT -m state --state INVALID -j DROP



# Blocked packets doesn't valids , marked by UNCLEAN
$IPT -A INPUT -m unclean -j LOG --log-level info
$IPT -A INPUT -m unclean -j DROP
# Blocked fragments
$IPT -A INPUT --fragment -j LOG --log-level info
$IPT -A INPUT --fragment -j DROP
# Blocked IP Address invalids
IPT -A INPUT -i eth0 -s 10.0.0.0/8 -j LOG_DROP_IPINVALIDO
$IPT -A INPUT -i eth0 -s 172.16.0.0/12 -j LOG_DROP_IPINVALIDO
$IPT -A INPUT -i eth0 -s 127.0.0.0/8 -j LOG_DROP_IPINVALIDO
# Blocked IP Address Multicast and Null Address
$IPT -A INPUT -i eth0 -s 255.255.255.255 -j LOG_DROP_IPINVALIDO
$IPT -A INPUT -i eth0 -s 0.0.0.0 -j LOG_DROP_IPINVALIDO
$IPT -A INPUT -i eth0 -s 224.0.0.0/4 -j LOG_DROP_IPINVALIDO

########
######## Specific Services to each Server and Firewall
########
# To HTTP Server
$IPT -A INPUT -p tcp --sport 1024: --dport 80 -m state --state \
NEW -j ACCEPT
# Accept SSH, only of 192.168.7.12
```

```
$IPT -A INPUT -s 192.168.7.12 -p tcp --sport 1024: --dport 22 -m state
-state \ NEW -j ACCEPT
# Disable connections  SSH for other address
$IPT -A INPUT -p tcp --dport 22 -j REJECT
# Accept SSH
$IPT -A INPUT -p tcp --sport 1024: --dport 22 -m state --state \
NEW -j ACCEPT
# To SMTP Server Accept POP3
$IPT -A INPUT -p tcp --sport 1024: --dport 110 -m state --state \
NEW -j ACCEPT
# To SMTP Server Accept IMAP
$IPT -A INPUT -p tcp --sport 1024: --dport 143 -m state --state \
NEW -j ACCEPT
# To SMTP Server
$IPT -A INPUT -p tcp --sport 1024: --dport 25 -m state --state \
NEW -j ACCEPT
# To DNS Server –53/tcp
#$IPT -A INPUT -p tcp --sport 1024: --dport 53 -m state --state \
#NEW -j ACCEPT
# To DNS Server -53/udp
$IPT -A INPUT -p udp --dport 53 -m state --state \
NEW -j ACCEPT
# Accept ICMP echo request
$IPT -A INPUT -p icmp --icmp-type echo-request -j ACCEPT
# OUTPUT Chain
# Let go out all the packets of the OUTPUT chain
$IPT -A OUTPUT -m state  --state NEW,ESTABLISHED,RELATED -j ACCEPT
# Queries DNS
$IPT -A OUTPUT -o eth0 -d $DNS_SERVER1 -p udp --dport 53 -m state \ -
state NEW  -j ACCEPT
#################################################################
#                    PROTECTION IN THE KERNEL
################################################################
# Ignore echo ignore broadcast
echo 1 > /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts
# Disable packets source routed
echo 0 > /proc/sys/net/ipv4/conf/all/accept_source_route
# Protection of SYN Cookie
echo 1 > /proc/sys/net/ipv4/tcp_syncookies
# Disable the acceptation of ICMP Redirect
```

```
echo 0 > /proc/sys/net/ipv4/conf/all/accept_redirects
# Disable the send of ICMP Redirect
echo 0 > /proc/sys/net/ipv4/conf/all/send_redirects
# Optimization Netfilter, accept 10000 connections
echo 10000 > /proc/sys/net/ipv4/ip_conntrack_max
```

3.7. Intrusion Detection System in the Security Design

We have installed three intrusion detection systems based in "**snort1.9.1"** over "**Red Hat 8.0"**. There are several IDS systems in the market, but we have selected snort because of its helpful support, which includes upgrade of signatures and versions, and because it is considered, by the security community, as the most important IDS system.

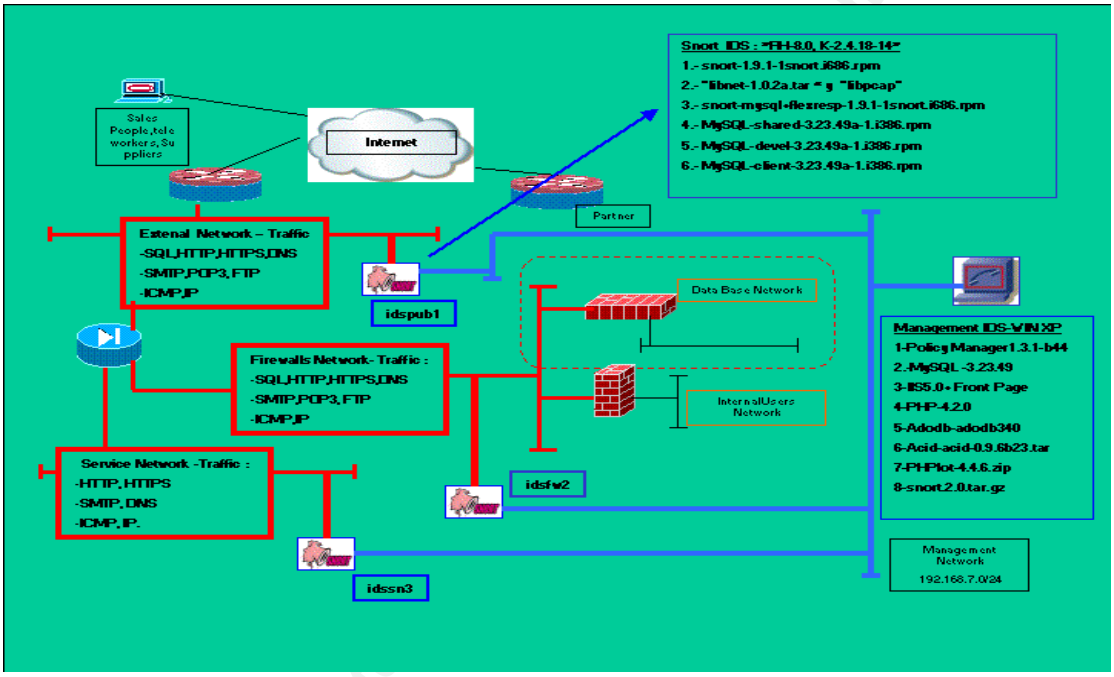


Figure 1 – IDS Architecture

This picture shows a topology of an distributed "ids" architecture. Each installed sensor has the following software packages:

- snort-1.9.1-1snort.i686.rpm
- libnet-1.0.2a.tar and libpcap.
- snort-mysql+flexresp-1.9.1-1snort.i686.rpm
- MySQL-shared-3.23.49a-1.i386.rpm
- MySQL-devel-3.23.49a-1.i386.rpm
- MySQL-client-3.23.49a-1.i386.rpm

The design considers to use a different sensor for each zone. The sensors are configured with different levels of security and traffic, according to its zone.

In the following table is described each component of the design:

Component	Description
idspub1	It is based in snort and has the responsibility of analyze the

	public network's traffic. In the public network we must configure the property of **"port monitor"** for these ports that are directly connected to this IDS, for example: the externals firewalls (2 ports) and the border router. So, our port monitor will receive the data of these 3 ports. The principal protocols analyzed by this IDS are: "**http,htttps,dns,smtp,ftp,icmp,ip; exploits".** ***Link to how configured the port monitor in Catalyst 2950.**
idsfw2	It is also based in snort. It has the responsibility of analyze the traffic of the network firewalls. We need to configure the port monitoring property in order to analyze the traffic of the main firewall's internal interface, the database firewall's main interface, and the internal users firewall. The principal protocols analyzed by this IDS are: "**sql, oracle, http, htttps, dns, smtp, ftp, icmp , ip; exploits.**
idssn3	It is also based in snort. It has the responsibility of analyze the traffic of the service network. We need to configure the port monitoring property in order to analyze the traffic of the following servers: web1, web2, mail, dns, and dmz_ser. The principal protocols analyzed by this IDS are: **"http, https, dns, smtp, oracle, sql, ftp, icmp , ip; exploits.**
Management	It is based in Policy Manager over "Windows XP". It will manage our ids system in a graphic way. In addition, it will receive and record all the log in a "MySQL" database. It will be also possible to view the recorded logs using a web page (over acid). To get this, we have used the suggestions given in the following sans room's practical guide: ***A Practical Guide to Running SNORT on Red Hat Linux 7.2 and Management Using IDS Policy Manager MySQL + IIS + ACID From your Workstation. By William Metcalf. http://rr.sans.org/intrusion/practial_guide.php*** This component was installed using the following packages : • Policy Manager : IDSPolMan-1.3.1.build44 • MySql : mysql-3.23.49-win.zip • Internet Information Service 5.0 and Front Page Server Extentions 2000. • PHP: php-4.2.0-installer.exe

	• Adodb: adodb340.tar • Acid : acid-0.9.6b23.tar • PHPlot : phplot4.4.6.zip • Snort: snort.2.0.tar.gz • Putty.

Tabla 1 – Summary

The following pictures show a summary of our IDS architecture. We are including an adequate policy to each sensor (all policy locations will be loaded from official: "snort.conf").

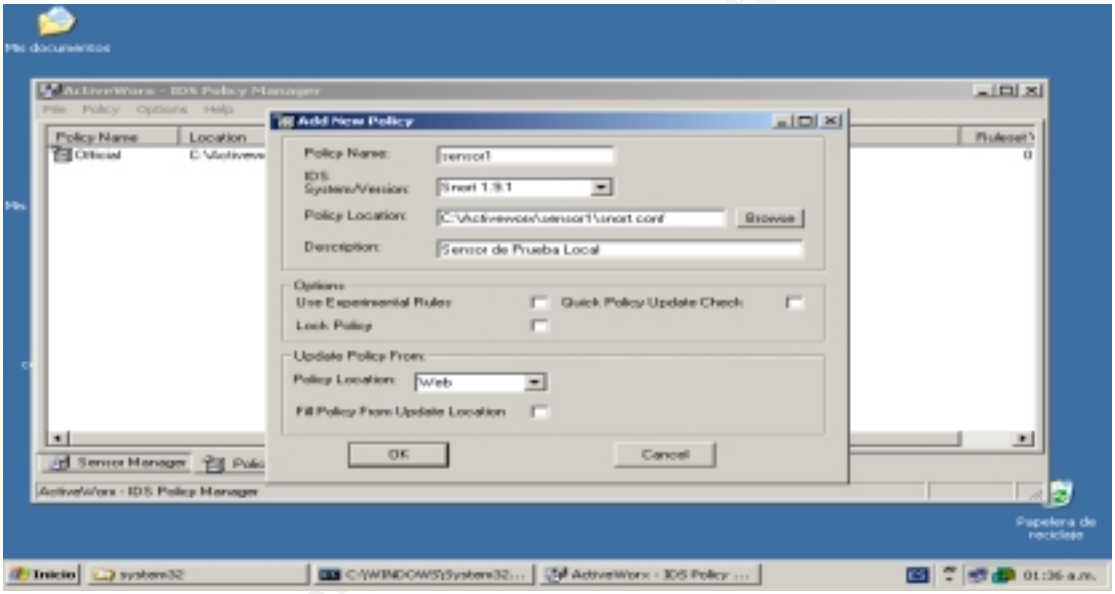


Figure 2 –**Add New Policy**

We have added an ids "sensor1" (this is the same for "idspub1"). Now, we could test the communication with this sensor using the SSH protocol (in our example we are using SCP):

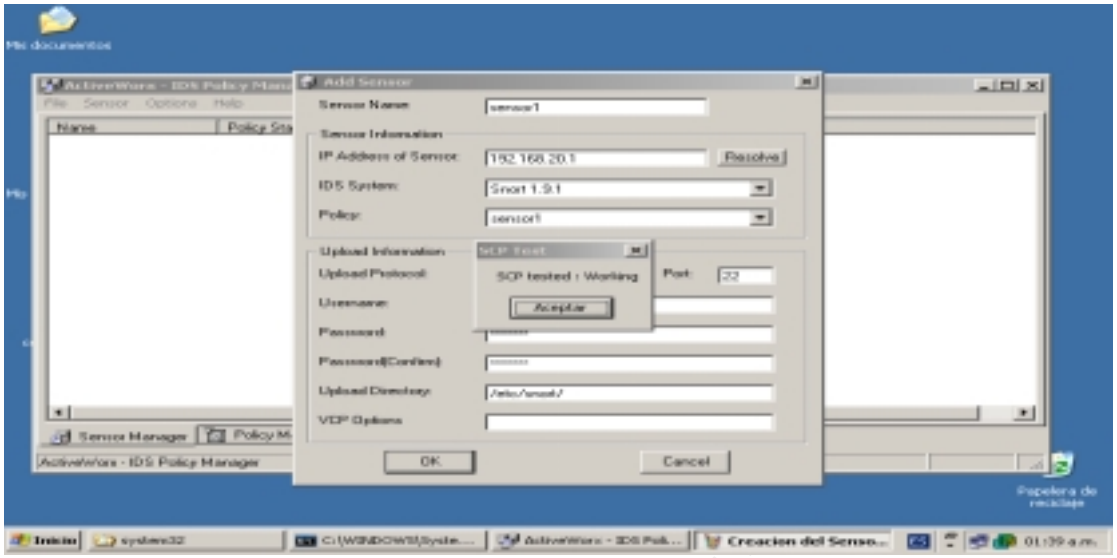


Figure 3- **SCP Tested**

The installation follows with an MySql server 3.23.49. After we have finished installing it, it is necessary to execute the option “WinMySQLAdmin” in order to create a database for snort:

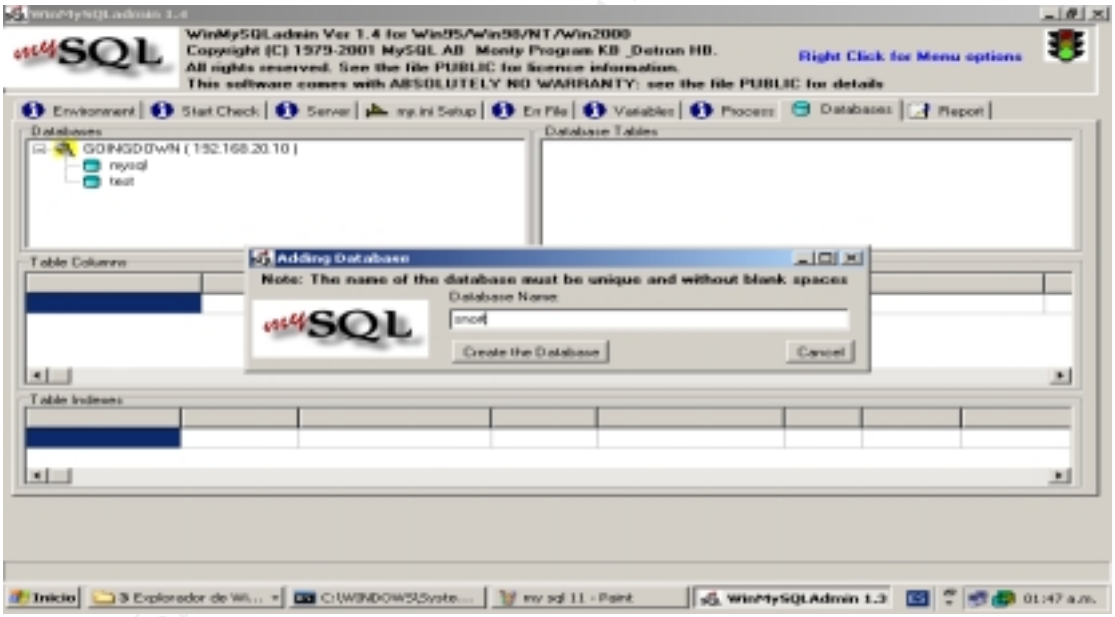


Figure 4 - Created the New Database

Then, we need to install PHP with support for IIS.

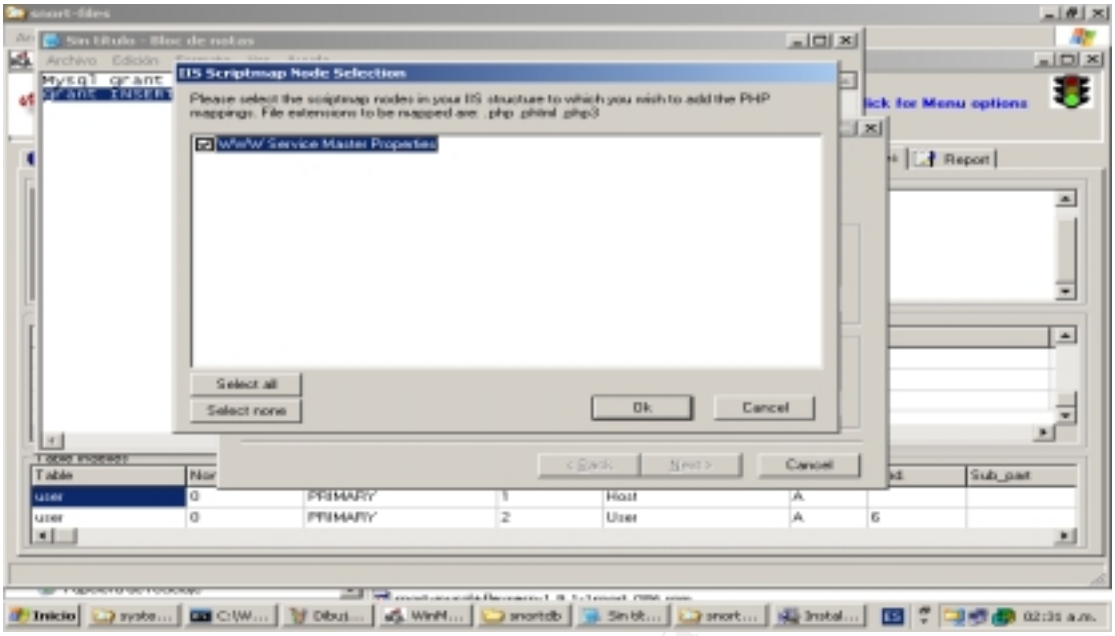


Figure 5- Supported to IIS

After that, please modify the following file: **"adodb.inc.php"**. This will enable to execute "ADODB" in an appropriate way.

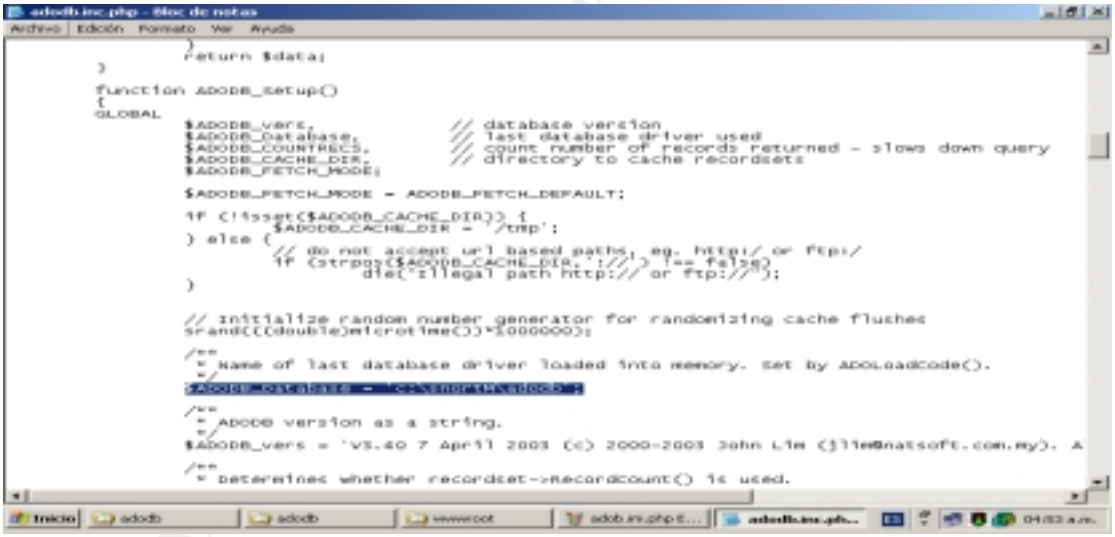


Figure 6- Modifying the file adoddb.inc.php

Also, it is necessary to modify the file **"acid_conf.php"** of the program **"acid"** (this file is located in "c:\inetpub\wwwroot\acid")**.**

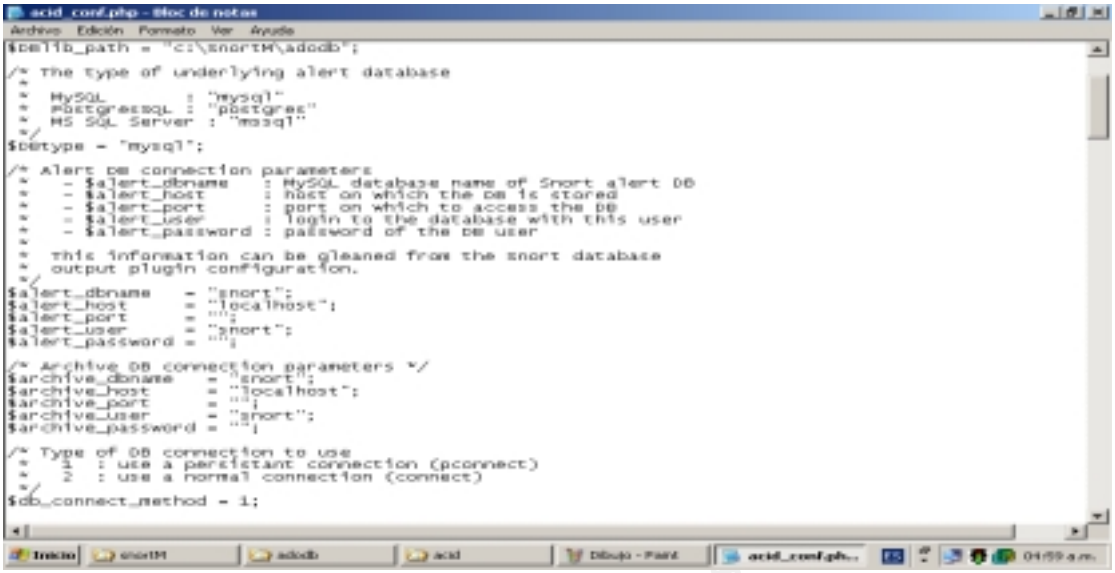


Figure 7- Modifying the file acid_conf.php

The name of the database and the IP address of the database server have to be filled (and checked) in the option “DataBase” of the “Policy Manager” section.

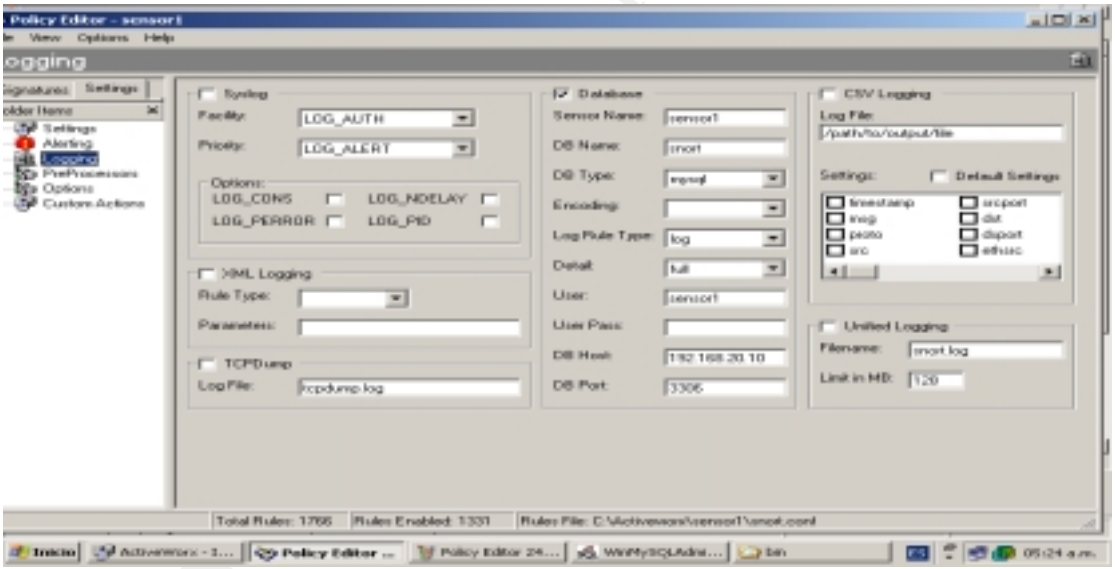


Figure 7- Enable the DB in the Policy Manager

Now, using a console window, it is necessary to execute a test of snort. The command need to use the option “-c” in order to specify which configuration file snort will use.

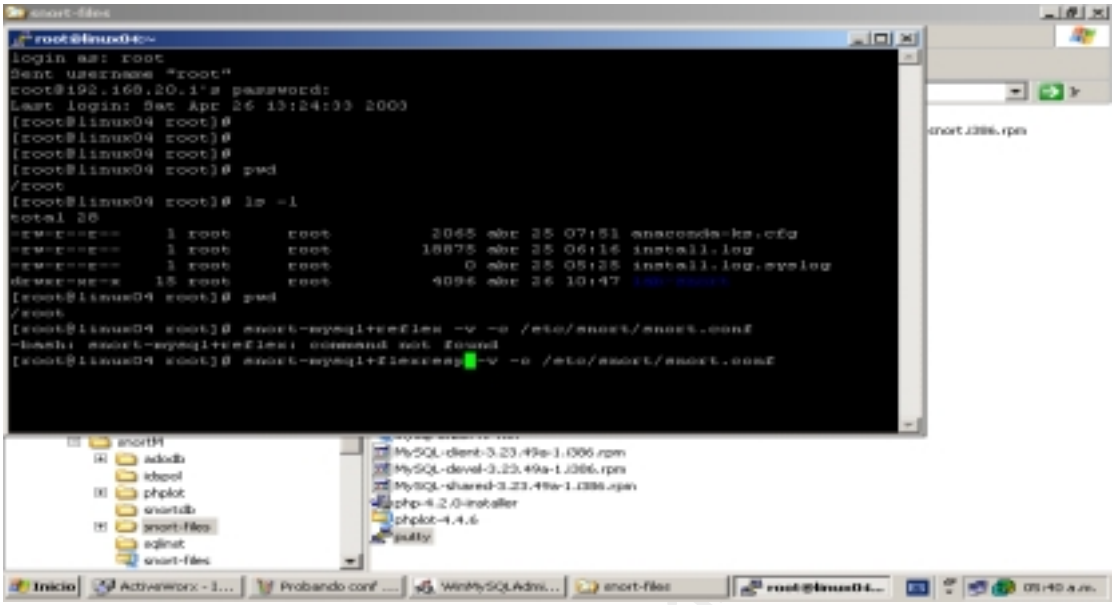


Figure 8- Running the snort

As an example, the following picture shows a typical raw of data (our example shows some netbios traffic):

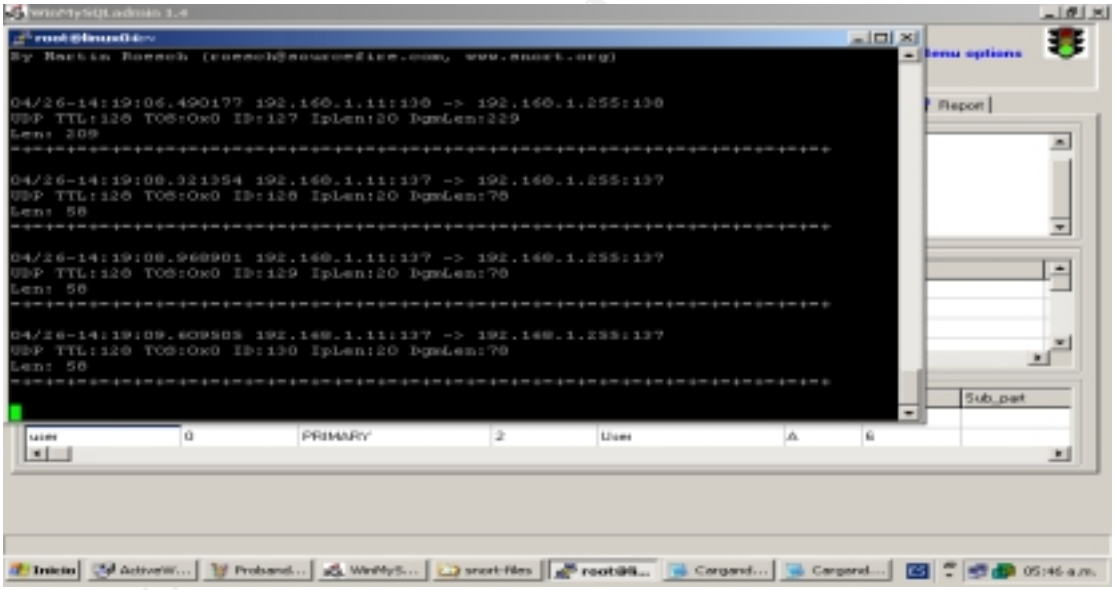


Figure 9 –test of capture data

In the following pictures it is showed the signatures that we will install for idspub1 (the IDS that will be installed in the public zone). This IDS needs to have enabled all its signatures because the public zone is where different types of traffic will exist.

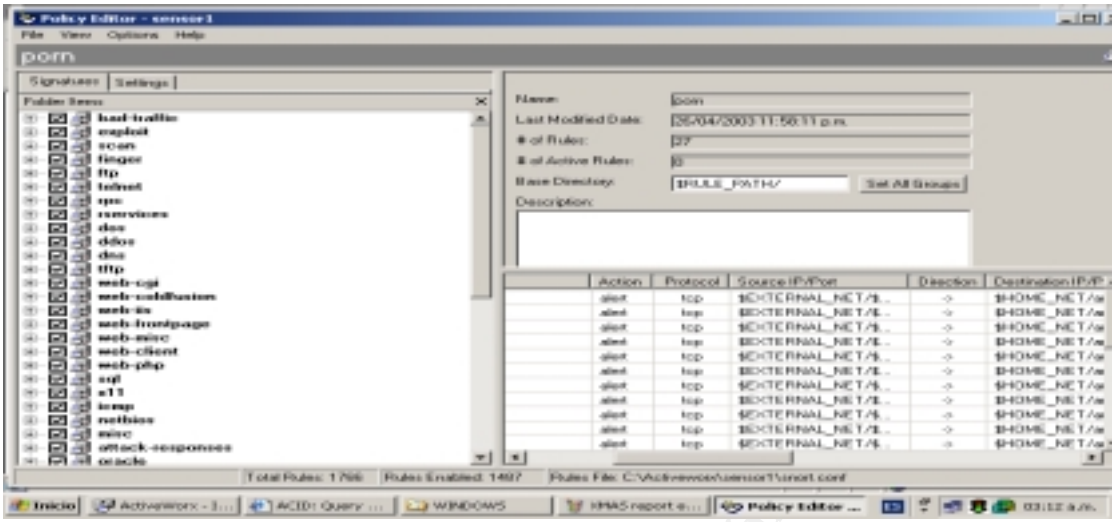


Figure 10 –Signatures of IDSpub1

The signatures of the internal IDS "idsfw2" is showed bellow. This IDS will analyze the traffic of the web server (and the database traffic), the internal users (internet request traffic), and internals servers like: dns, smtp (all of them with their respective external dns and smtp).

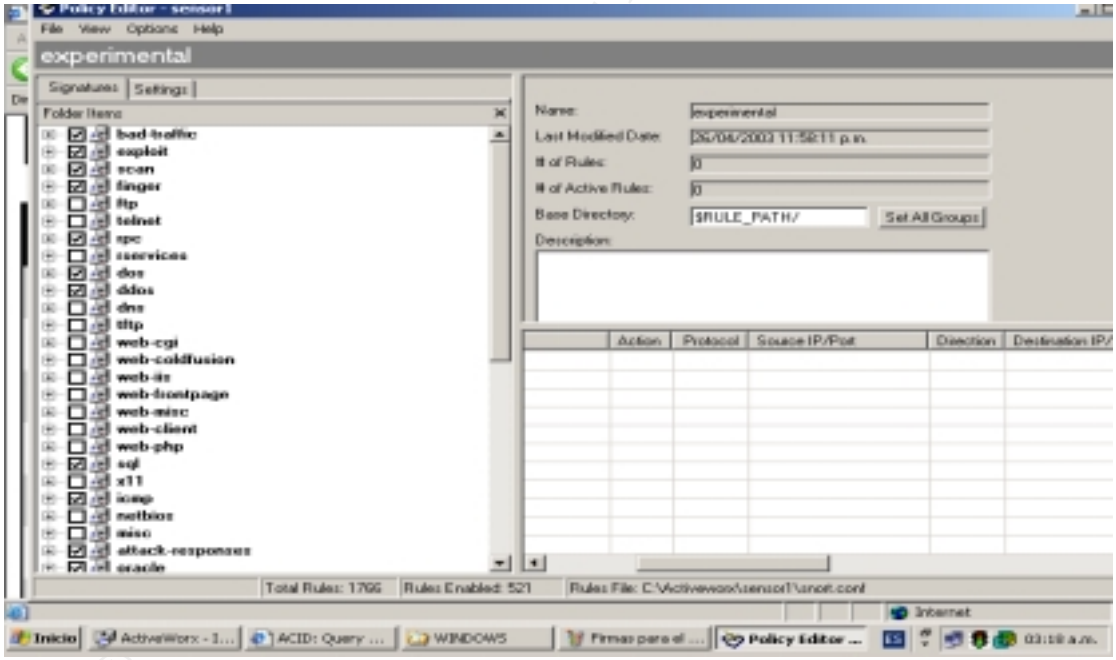


Figure 11- Signatures of IDSfw2

In order to analyze the oracle database traffic, it is necessary to check the oracle signature. This is showed in the following picture:

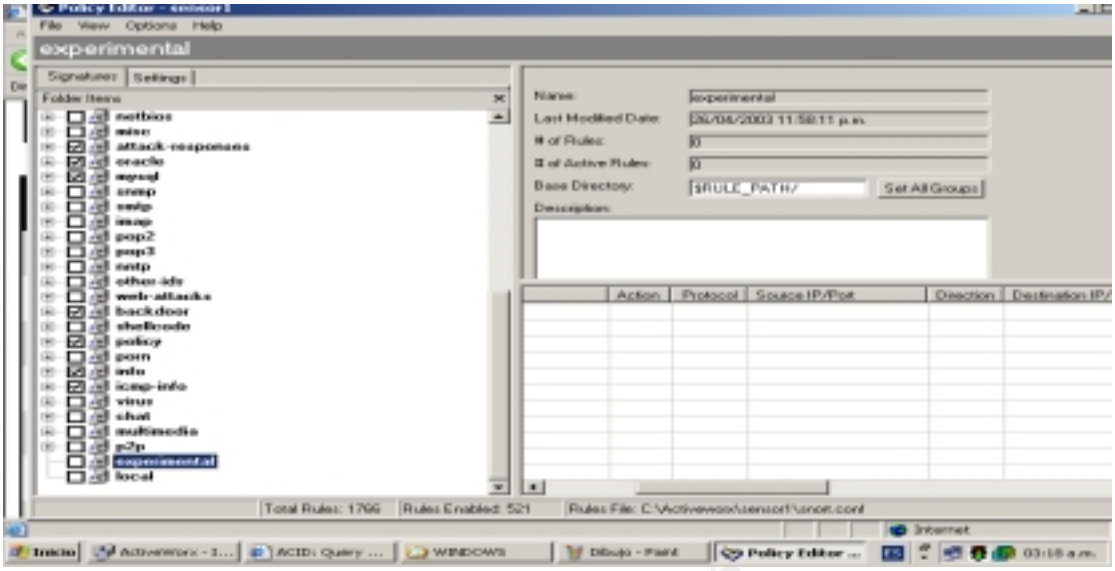


Figure 12 - Signatures of idsfw2

Finally, we need to configure the IDS for the service network. This IDS will analyze the traffic of: web, dns, and smtp server.

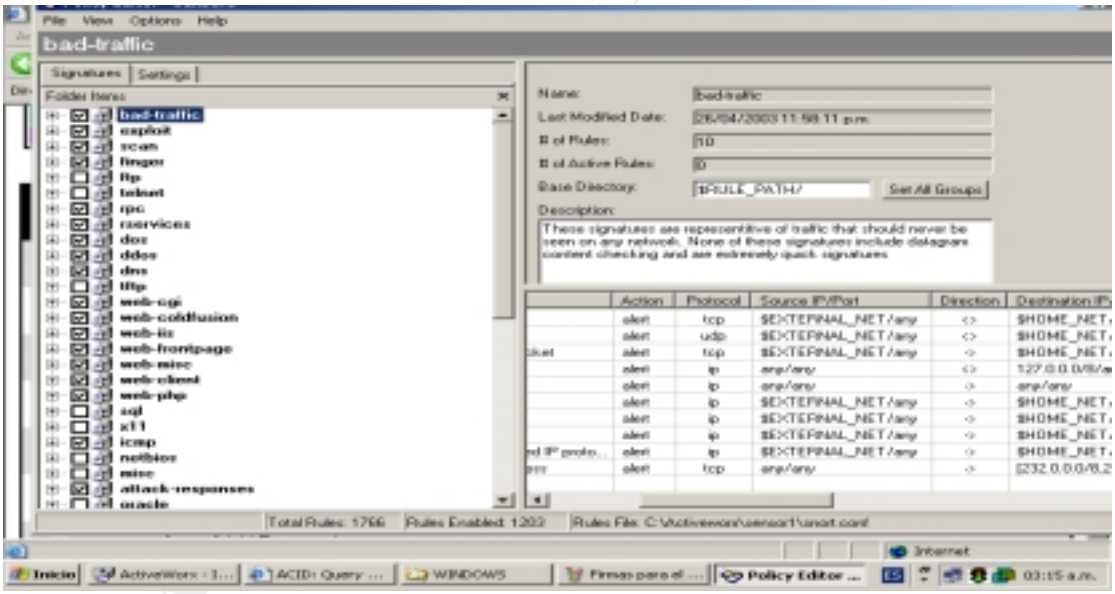


Figure 13- Signatures of IDSsn3

In the Signatures section it will be checked the following options: **"attack-response"** , **"web-attacks**", **"back-door".** Of course, it will also be necessary to check our origin signatures.

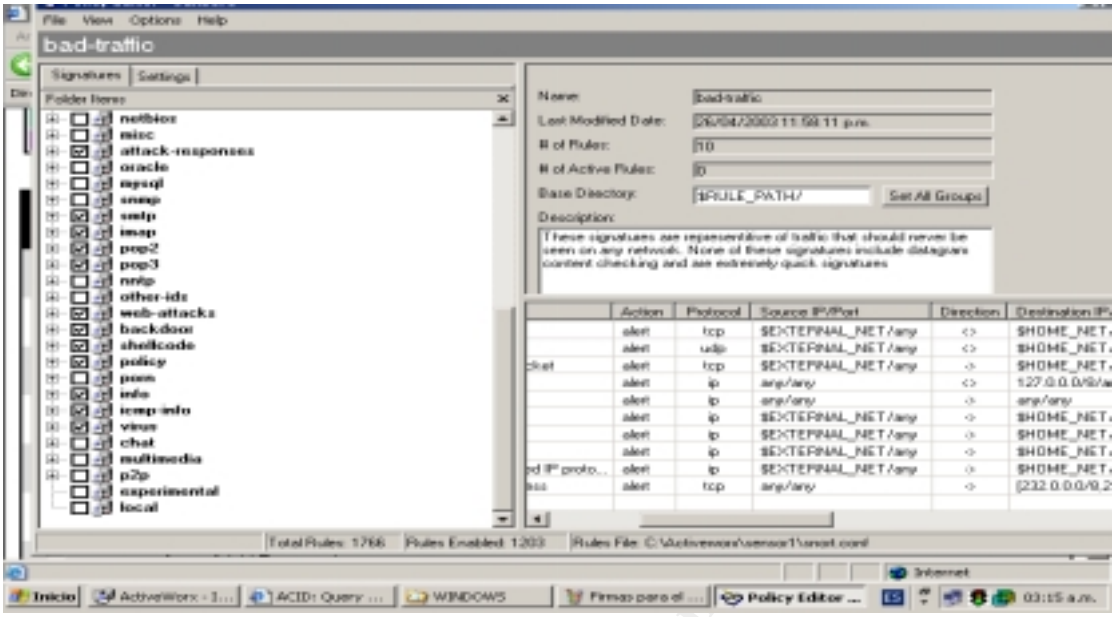


Figure 14- Signatures of IDSsn3

The configuration of "preprocessors" is showed bellow. These items include: Stream Reassembly , Stream4, HTTP Decode, RPC Decode, frag2, etc.

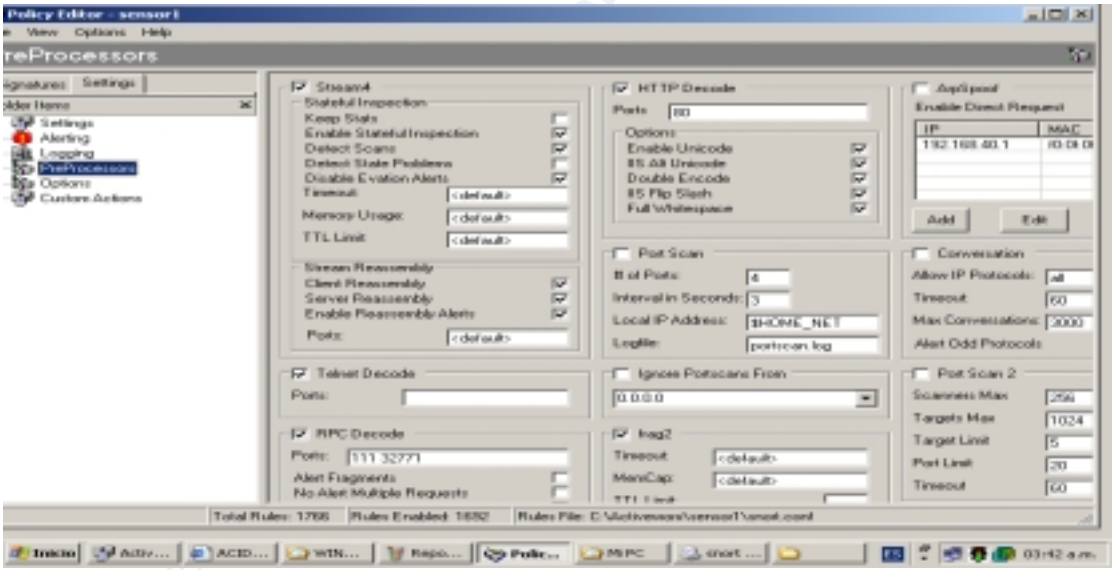


Figure 15- **PreProcessors**

Now, with the policy configured, we can install the policy to each sensor (using the "Policy Editor" option).

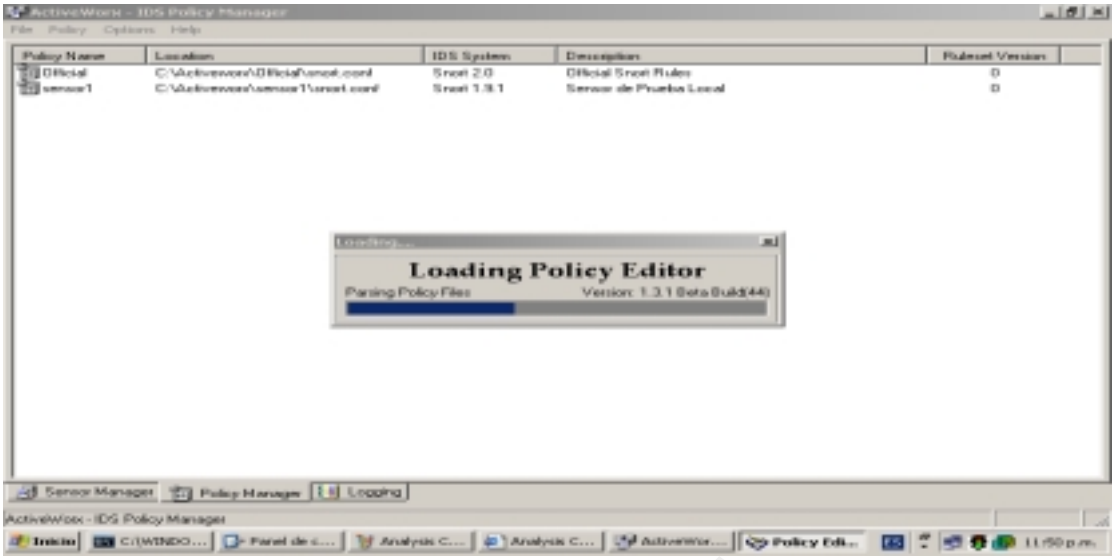


Figure 16- Loading the Policy

Finally, the following two pictures show the kind of reports that we can get.

In our picture, these reports show a summary of a sample traffic:

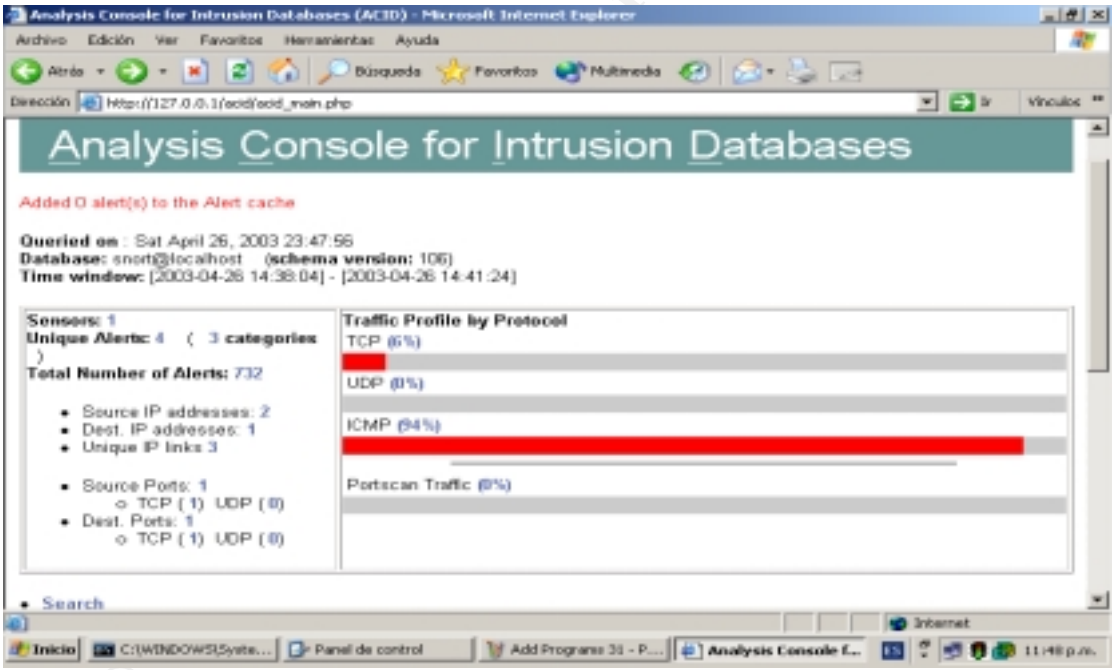


Figure 17- Analysis the Console via http

The last picture shows the traffic recorded between a computer running **"nmap "** (with the option XMAS Scan) and a victim host running linux installation by default.

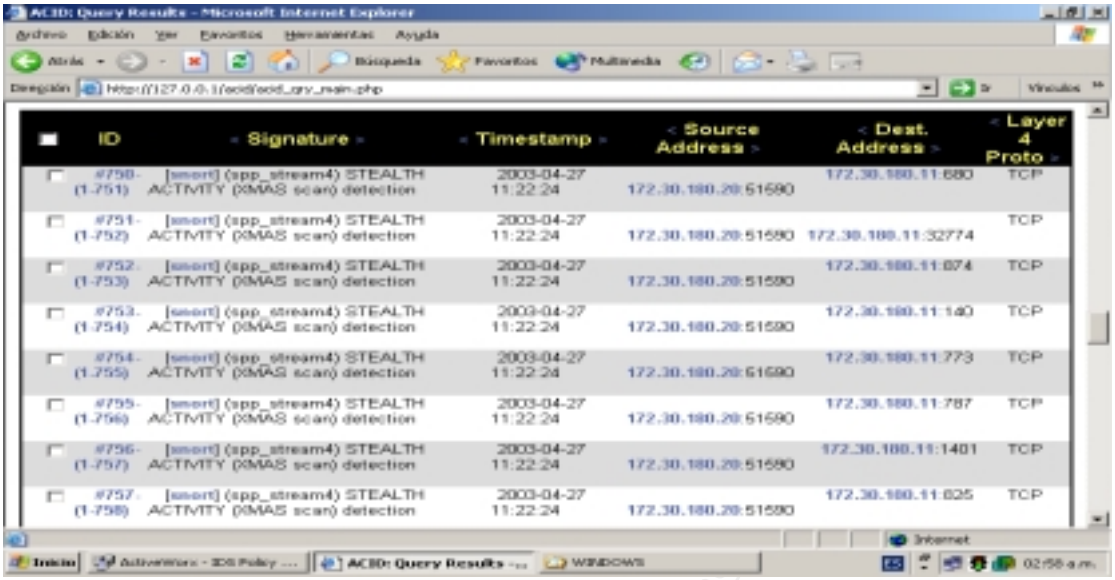


Figure 18- Analysis the Console via http

4. ASSIGNMENT 3 – AUDIT SECURITY INFRASTRUCTURE

4.1. Primary Firewall Audit Plan

4.1.1. Introduction

The goal of the audit is to ensure that undesired access is not permitted according to the security policies implemented in the network.

The audit might be done after the initial implementation and after any modification to the rules of the firewall, that ensures that the change doesn´t break any other application functions.

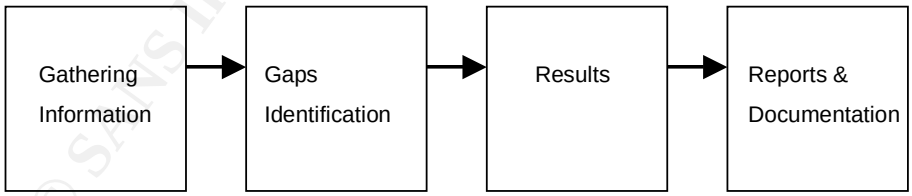
Routine testing of networks can greatly reduce the chances of a network compromise by helping to ensure the critical systems.

The scope of this audit is the main firewall, if it is working like its rules said and if by itself it is protected.

Following the principle of "separation of duties", GIAC Enterprises give the audit responsibility to another security group, they have not been involved in the design or implementation of the network security.

4.1.2. Methodology

We have decide to use the next methodology:



In order to do that, we will follow the next steps:

- First, test if it is possible to gather some information of the firewall that can be used to launch an attack to the network.

- Second, test if the firewall is vulnerable to non-standard TCP/IP traffic that can help an attacker to bypass the security policy or get some information of the internal network.
- Third, test if the firewall works like its security policy says. That is, permit only the specific traffic we have configured and deny the rest.

After all the phases, the results should be documented and made available for staff and security groups.

4.1.3. Audit information

In order to avoid unnecessary risks, GIAC Enterprises decided to limit the servers that will be part of the tests to the web server running HTTP and HTTPS.

Also, all the tests will be done from an external perspective, in the outside LAN and/or from Internet.

4.1.4. Risks and Considerations

In order to perform auditing activities, aspects as maintenance window should be taken carefully into account. The actual audit will be conducted on one of the weekends. The test will start from Saturday morning 6:00am, and is expected to finished by Sunday morning within 24 hours.

All the systems in the architecture are fullly backed up by the technical personnel before the perform the audit.

4.1.5. Cost and Effort Level

In our case, the cost for the software and hardware for the audit is minimal. GIAC Enterprises will use open source software tools for the testing and vulnerability analysis and the hardware will be two laptops used in the administration of our networks.

The time estimated to do all the audit work is 30 hours separated in three days, beginning at Friday afternoon (GIAC Enterprises will put all their services in maintenance state) and finishing at Sunday late in the afternoon.

The cost per hour of the personnel is:

Resource	Qty	Cost P/H
Security Audit	1	100

Network Support	1	80
Security Support	1	90
Total Cost (US$)		**8100**

4.1.6. Documentation

At the end of the audit work the security group in charge must present a security report with the results and all the issues they found in their work.

4.1.7. Tools

NMAP (http://www.insecure.org/nmap)

Nmap ("network mapper") is an open source utility for network exploration or security auditing. It was designed to rapidly scan large networks, although it also works against single hosts. Nmap uses raw IP packets to determine what hosts are available on the network, what services (ports) they are offering, what operating systems (and version) they are running, what type of packet filters/firewalls are in use, and other characteristics

The command line format for running nmap is as follows:

nmap [scan type (s)] [options] <hosts or nets>

An example SYN scan of a class C network is shown:

nmap –sS –P0 –v –O –p 1-12000 –oN scan.txt 200.20.YY.x/24

where:

- -sS: SYN scan
- -P0: Do not ping
- -v: verbose mode
- -O: fingerprint OS
- -p 1-12000: ports 1 to 12000
- -oN scan.txt: log results in a human readable format to scan.txt
- 200.20.YY.x/24: the subnet

Nessus (http://www.nessus.org)

The premier Open Source vulnerability assessment tool Nessus is a remote security scanner for Linux, BSD, Solaris, and other Unices. It is plug-in-based, has a GTK interface, and performs over 1200 remote security checks. It allows for reports to be generated in HTML, XML, LaTeX, and ASCII text, and suggests solutions for security problems.

HPING: http://www.hping.org

A network probing utility like ping on steroids hping2 assembles and sends custom ICMP/UDP/TCP packets and displays any replies. It was inspired by the ping command, but offers far more control over the probes sent. It also has a handy traceroute mode and supports IP fragmentation. This tool is particularly useful when trying to traceroute/ping/probe hosts behind a firewall that blocks attempts using the standard utilities. The syntax GIAC is going to use are as following:

hping2 -V --frag --data 150 –count 10 --syn -p 80 ip_address

-*V* verbose

--*frag* fragmentation

--*data* the size of the data

--*count* how many packets we are going to send

--*syn* set the flag

-p 80 scan port 80

ip_address target address

TCPdump (http://www.tcpdump.org)

TCPdump is a powerful tool for network sniffering. This program allows you to see the traffic on a network. It can be used to print out the headers of packets on a network interface that matches a given expression. You can use this tool to track down network problems, to detect attacks or to monitor network activities.

4.1.8. Auditing Diagram

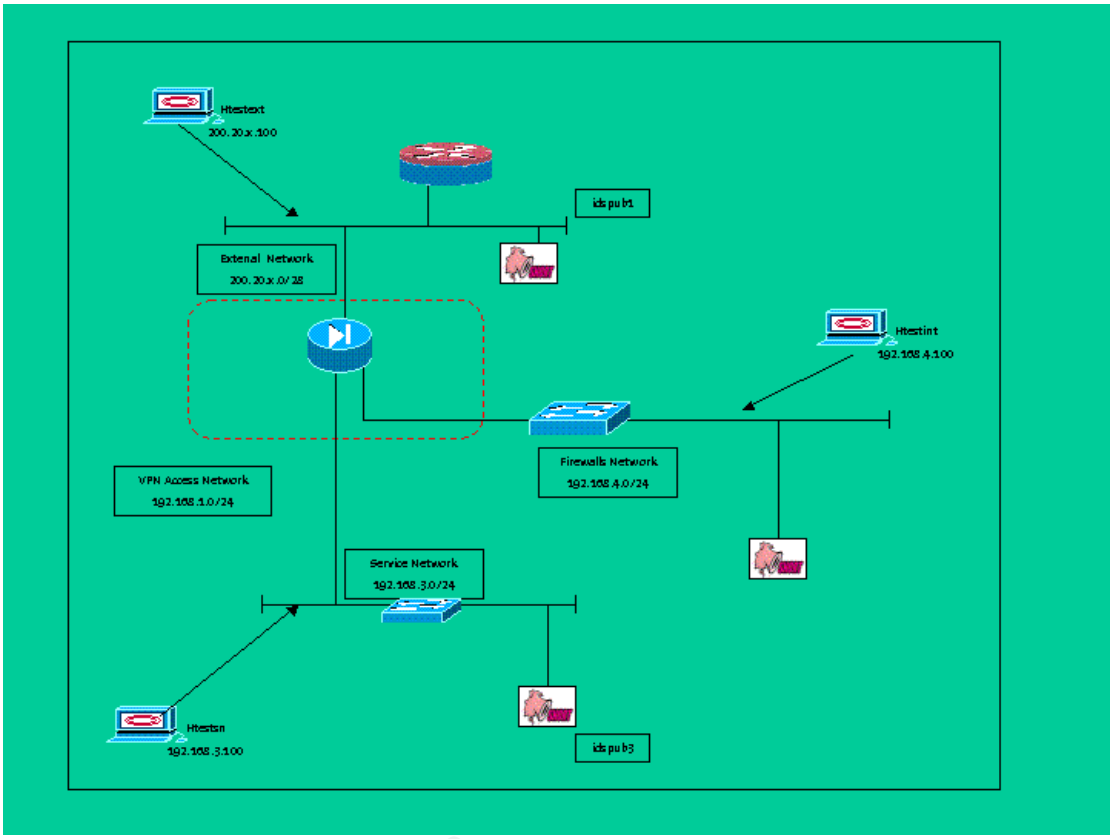


Figure 1 – Auditing Diagram

4.1.9. Executions

4.1.10. Test if we can find the "the firewall" and what type of firewall is.

Command:

nmap –v –g500 –sS –sR –P0 –O 200.20.x.2

Explanation of the command:

Testing the firewall in order to find some information that can help us determined which firewall we are using. We choose some options that are specific for this type of application.

Options:

- -v: verbose mode
- -g500: using a source port, in this case we suppose that the firewall can be used as as VPN concentrator (UDP 500 isakmp)
- -sS: use TCP scan
- -sR: try to find RPC ports in the open ports it detects.
- -O: fingerprint OS detection

- -P0: do not try and ping hosts. It permits scanning of networks that filter ICMP.
- 200.20.X.2: the IP of the firewall

Results:

According to the results of the Nmap, we can find that there is a firewall protecting the network ("filtered ports"), but the fingerprint OS detection is unable to detect which firewall are we using.

```
root@dogbert:~
File  Edit  View  Terminal  Go  Help
[root@dogbert root]# nmap -v -g500 -sR -sS -O -P0 200.20.X.2

Starting nmap V. 3.00 ( www.insecure.org/nmap/ )
Host  (200.20.X.2) appears to be up ... good.
Initiating SYN Stealth Scan against  (200.20.x.2)
The SYN Stealth Scan took 1733 seconds to scan 1601 ports.
Initiating RPCGrind Scan against  (200.20.x.2)
The RPCGrind Scan took 0 seconds to scan 0 ports.
Warning:  OS detection will be MUCH less reliable because we did not find at least 1
open and 1 closed TCP port
All 1601 scanned ports on  (200.20.x.2) are: filtered
Too many fingerprints match this host for me to give an accurate OS guess
TCP/IP fingerprint:
SInfo(V=3.00%P=i686-pc-linux-gnu%D=6/30%Time=3F00FAD4%O=-1%C=-1)
T5(Resp=N)
T6(Resp=N)
T7(Resp=N)
PU(Resp=N)



Nmap run completed -- 1 IP address (1 host up) scanned in 1947 seconds
[root@dogbert root]#
```

Our comments respect to this are:

- The NMAP can not recognize the PIX by the fingerprint OS detection feature. This means that it is more difficult to an attacker to exploit some vulnerability in the firewall.
- From the point of view of an attacker, it could be better if the output shows "closed ports" instead of "filtered ports" because this show us that there is an equipment that filters ports, i.e. a firewall.

  This could be done by changing the answers to ICMP or TCP resets.

The logs of the Cisco shows this:

```
pixfirewall#
pixfirewall# 402106: Rec'd packet not an IPSEC packet. (ip)
dest_addr= 200.20.x.2, src_addr= p402106:
Rec'd packet not an IPSEC packet. (ip) dest_addr= 200.20.x.2,
src_addr= 200.20.x.100p302010: 0 in use, 0 most used
```

Although this type of logs does not show an specific scan or attack it might be known by the security personnel in order to identify some scan to the firewall.

4.1.11. Test of the non-standard issues

These tests are done because some attackers uses the craft packets to bypass the firewall filters and gather information about hosts in the network.

4.1.11.1. External FIN Scan

Command:

nmap –v –g500 –sF –P0 –p 1-65535 200.20.x.3

Explanation of the command:

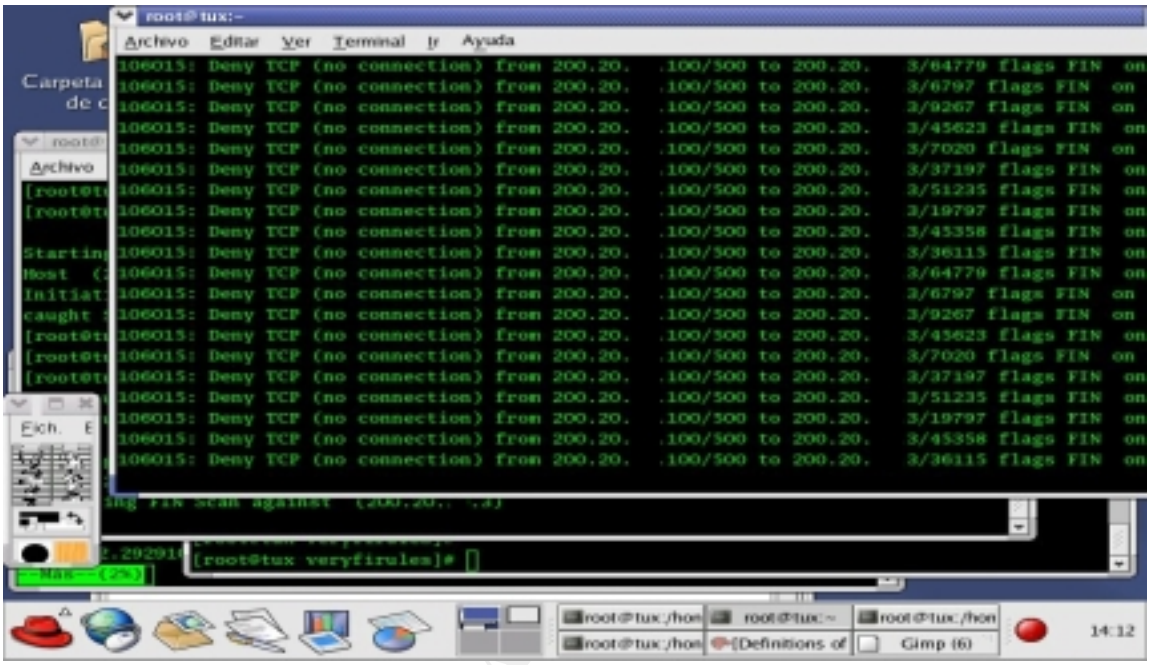
In this case we are sending a packet with the FIN flag set.

The options:

- -v: verbose
- -g500: using a source port
- -sF: Stealth FIN
- -P0: do not try ping hosts
- -p1-65535: test all the ports

Results:

The Firewall Log output shows : “Deny TCP (no connection)”



The Tcpdump output shows no answer from the firewall, which is good

```
11:52:39.148977  200.20.x.100.isakmp  >  200.20.x.3.57852:  F  0:0(0)  win
1024

11:52:39.150150 200.20.x.100.isakmp > 200.20.x.3.925: F 0:0(0) win 1024

11:52:39.150316  200.20.x.100.isakmp  >  200.20.x3.16358:  F  0:0(0)  win
1024

11:52:39.150781  200.20.x.100.isakmp  >  200.20.x.3.38673:  F  0:0(0)  win
1024

11:52:39.150927  200.20.x.100.isakmp  >  200.20.x.3.61119:  F  0:0(0)  win
1024

11:53:03.188454  200.20.x.100.isakmp  >  200.20.x.3.2390:  F  0:0(0)  win
1024
```

4.1.11.2. External Xmas Scan

Command:

nmap –v –g500 –sX –P0 –p 1-65535 200.20.x.3

Explanation of the command:

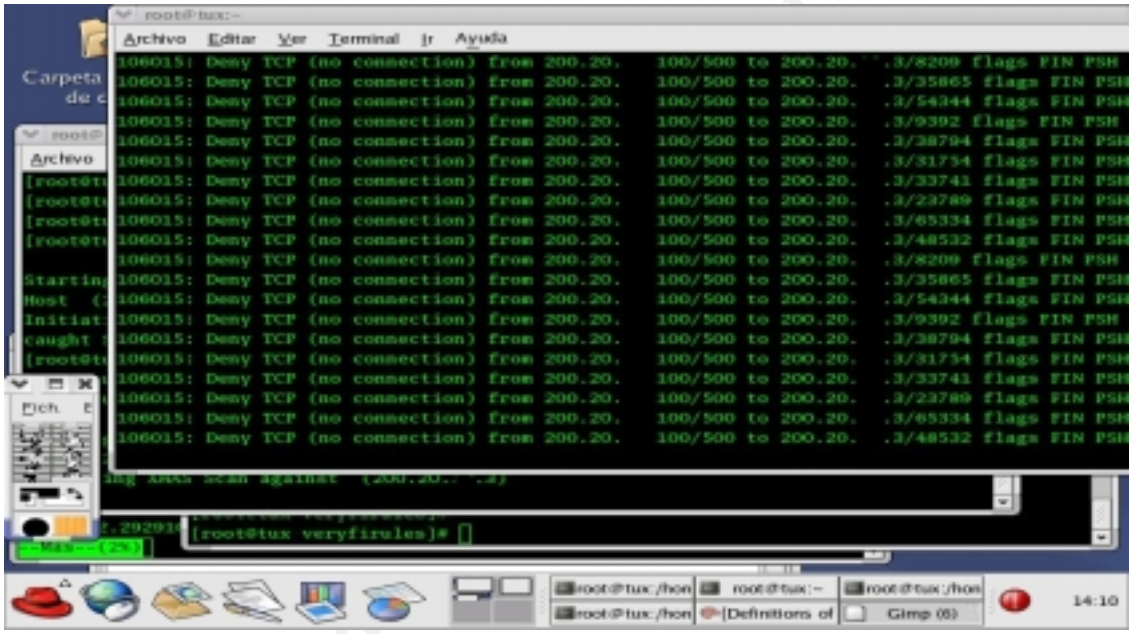
The same that above but in this case we send a Xmas tree test.

Results:

There is no answer from the firewall as shown in the NMAP output.

```
[root@tux veryfirules]# nmap -v -g500 -R -sX -P0 -p 1-65535
200.20.x.3
Starting nmap V. 3.00 ( www.insecure.org/nmap/ )
Host (200.20.x.3) appears to be up ... good.
Initiating XMAS Scan against (200.20.x.3)
```

The Firewall Log output shows : “Deny TCP (no connection)”



There is no answer from the firewall as shown in the Tcpdump output.

```
14:03:09.076885 200.20.x.100.isakmp > 200.20.x.3.56402: FP 0:0(0) win
2048 urg 0
14:03:09.076967 200.20.x.100.isakmp > 200.20.x.3.55233: FP 0:0(0) win
2048 urg 0
14:03:15.087954 200.20.x.100.isakmp > 200.20.x.3.33521: FP 0:0(0) win
2048 urg 0
14:03:15.088033 200.20.x.100.isakmp > 200.20.x.3.17197: FP 0:0(0) win
2048 urg 0
```

```
root@tux  veryfirules]#  nmap  -v  -g500  -R  -sN    -P0  -p  1-65535
200.20.x.3
```


4.1.11.3. External Null Scan

Command:

nmap –v –g500 –sN –P0 –p 1-65535 200.20.x.3

Explanation of the command:

We try a NULL Scan.

Results:

No answer from the firewall, look at the output.

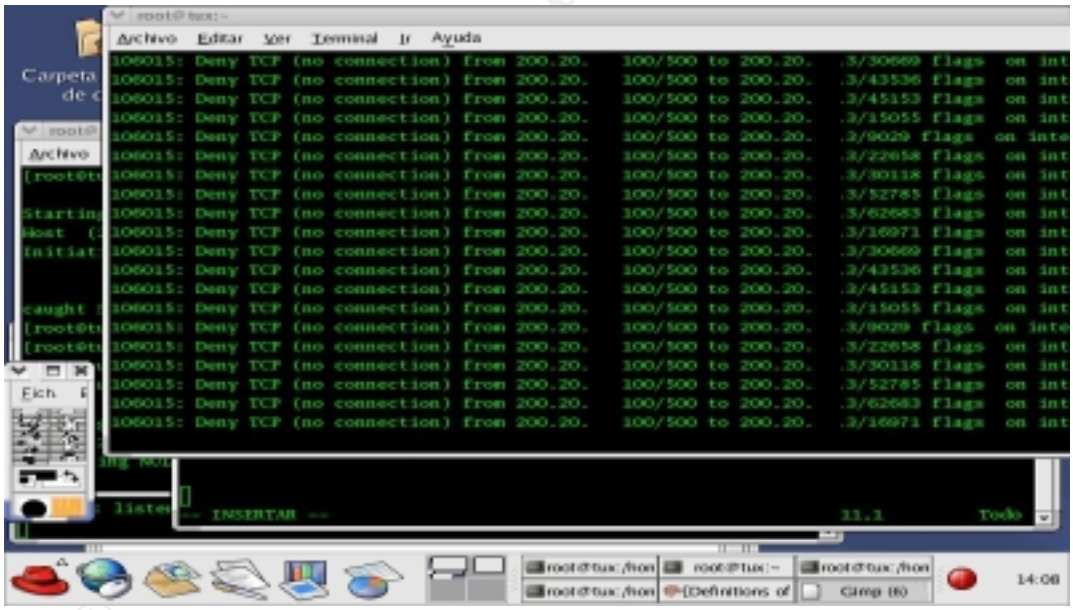
```
[root@tux  veryfirules]#  nmap  -v  -g500  -R  -sN    -P0  -p  1-65535
200.20.x.3

Starting nmap V. 3.00 ( www.insecure.org/nmap/ )

Host  (200.20.x.3) appears to be up ... good.

Initiating Null Scan against  (200.20.x.3)
Nmap run completed -- 1 IP address (1 host up) scanned
```

The Firewall Log output shows : “Deny TCP (no connection)”



There is no answer from the firewall as shown in the Tcpdump output.

```
14:04:12.939359 200.20.x.100.isakmp > 200.20.x.3.60147:  . win 1024
14:04:12.940622 200.20.x.100.isakmp > 200.20.x.3.12107:  . win 1024
14:04:12.940810 200.20.x.100.isakmp > 200.20.x.3.56257:  . win 1024
14:04:12.940957 200.20.x.100.isakmp > 200.20.x.3.19166:  . win 1024
```

4.1.11.4. External ACK scan

Command:

nmap –v –g500 –sA –P0 –p 1-65535 200.20.x.3

Explanation of the command:

We send packets with the ACK flag set to gather info from the server.

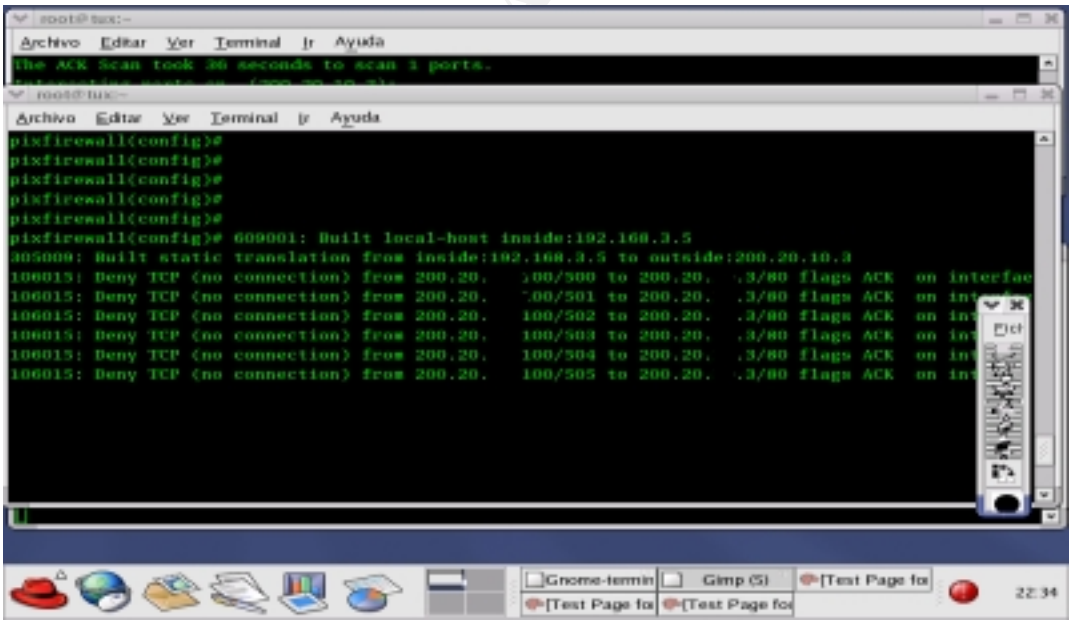
Results:

No answer from the firewall

```
[root@tux veryfirules]# nmap -v -g500 -R  -sA  -P0    -p 1-65535  200.20.x.3

Starting nmap V. 3.00 ( www.insecure.org/nmap/ )
Host  (200.20.x.3) appears to be up ... good.
Initiating ACK Scan against  (200.20.x.3)
```

The Firewall Log output shows : “Deny TCP (no connection)”



```
14:18:38.944307   200.20.x.100.isakmp   >   200.20.x.3.30798:   .   ack
1875241526 win 2048

14:18:38.946733   200.20.x.100.isakmp   >   200.20.x.3.44587:   .   ack
1875241526 win 2048

14:18:38.946999   200.20.x.100.isakmp   >   200.20.x.3.30916:   .   ack
1875241526 win 2048

14:18:38.947187 200.20.x.100.isakmp > 200.20.x.3.4482: . ack 1875241526
win 2048

14:18:38.947742   200.20.x.100.isakmp   >   200.20.x.3.48380:   .   ack
1875241526 win 2048
```

From the NMAP documentation:

The idea is that closed ports are required to reply to your probe packet with an RST, while open ports must ignore the packets in question (see RFC 794 pp 64). The FIN scan uses a bare FIN packet as the probe, while the Xmas tree scan turn on the FIN, URG, and PUSH flags. The Null scan turns off all flags. Unfortunately Microsoft decided to completely ignore the standard and do things their own way. Thus this scan type will not work against systems running Windows9S/NT.

4.1.12. Fragmentation effects from Internet

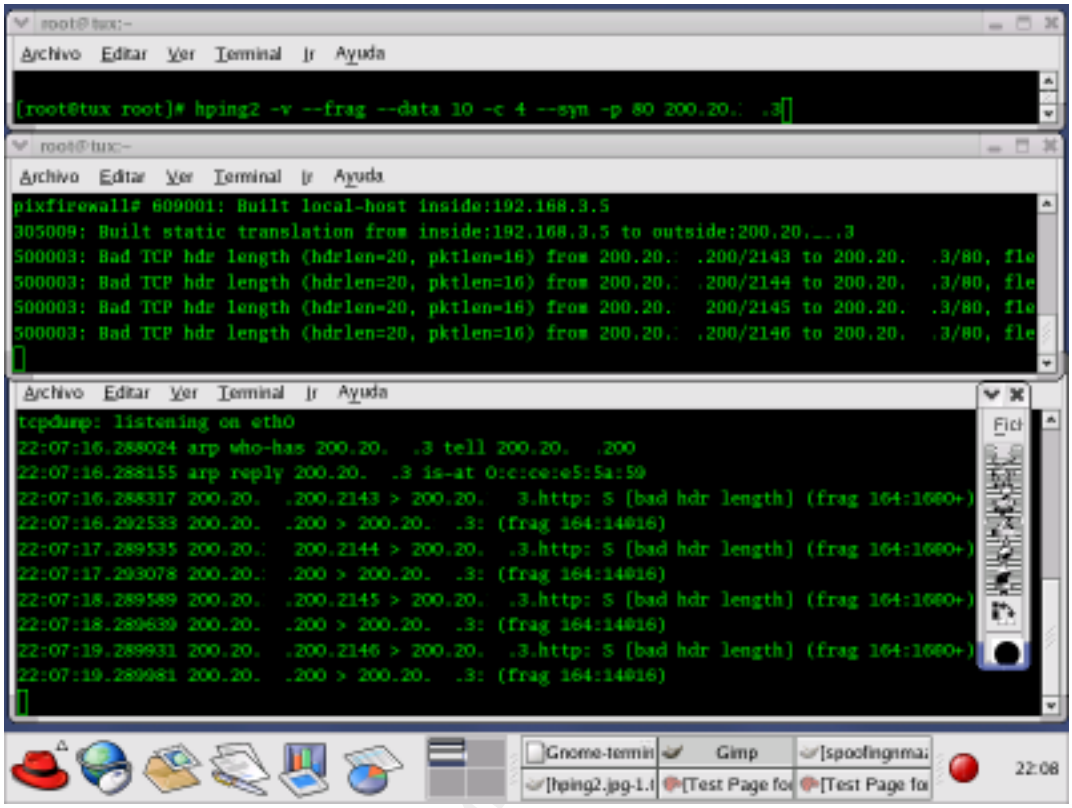
We will send some fragmented packets to port 80 on the web server. to see if the fragmentized packets could pass through the firewall.

Command:

hping2 –v --frag --data 10 –c 4 -sync -p 80 200.20.x.3

Explanation of the command:

We send packets fragmented with a data payload of 10 bytes to web server.



Cisco logs:

%PIX-5-500003: Bad TCP hdr length (hdrlen= bytes, pktlen= bytes) from src_addr/ sport to dest_addr/ dport, flags: tcp_flags, on interface int_name

Comments:

The Cisco PIX receives the fragment packet but because the payload is so short it indicates a violation of the TCP length header. The tcpdump shows no response from the web server.

The PIX can enforce the fragment attacks using the Frag Guard feature but it is not enable by default in the firewall.

4.1.13. Test the security policy rules

Remember that the web server is the only one connected to the network (for our testing purposes).

Command:

nmap -v -sS –P0 –p 1-65535 -oN test1.txt 200.20.x .0/24

Explanation of the command:

In this case we want to test all the network segment and all the possible privileged ports.

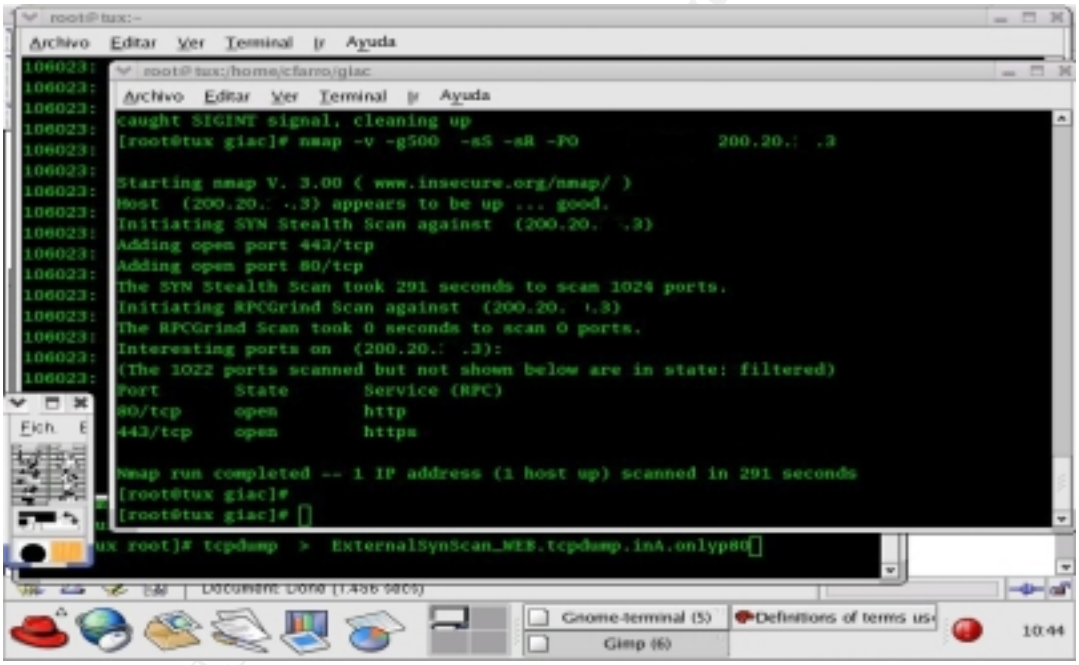
The options:

- -v: verbose mode
- -sS: TCP syn scan
- -p 1-65535: test all the ports
- -oN: output to a file in human readable format
- 200.20.x.3: the target hosts

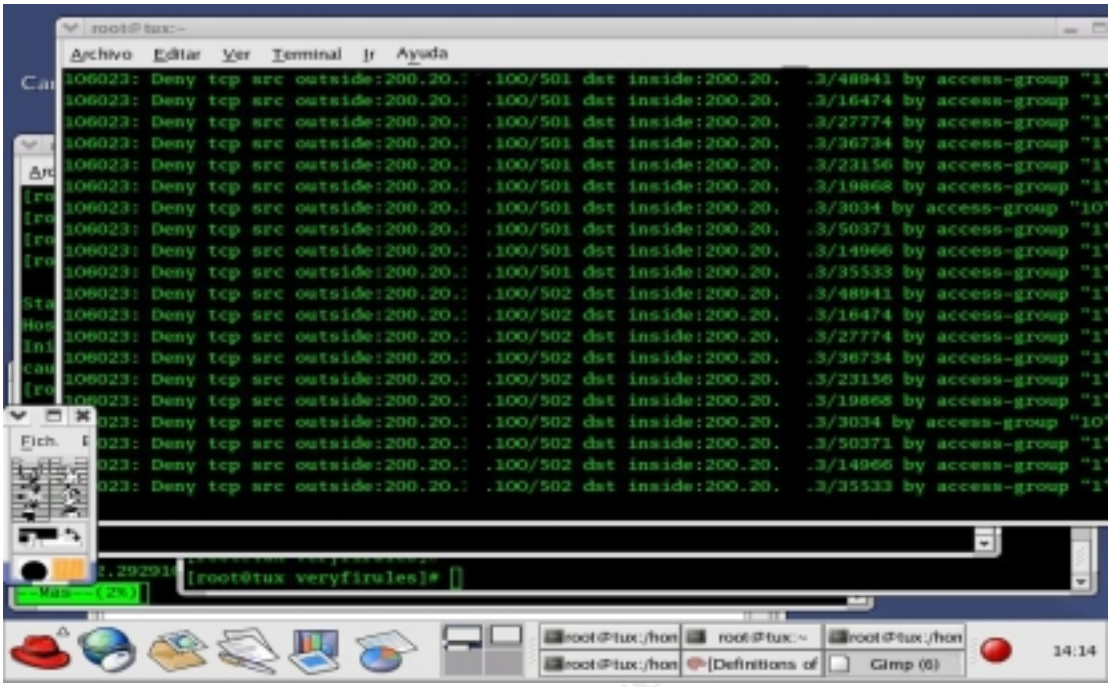
Results:

The nmap scan shows that the only open ports are: http (TCP port 80) and https (TCP port 443), which is what we have configured at the firewall.

The next screenshot shows the output for the nmap scan:



The output at Cisco logs shows all the deny traffic which corresponds to all the other ports.



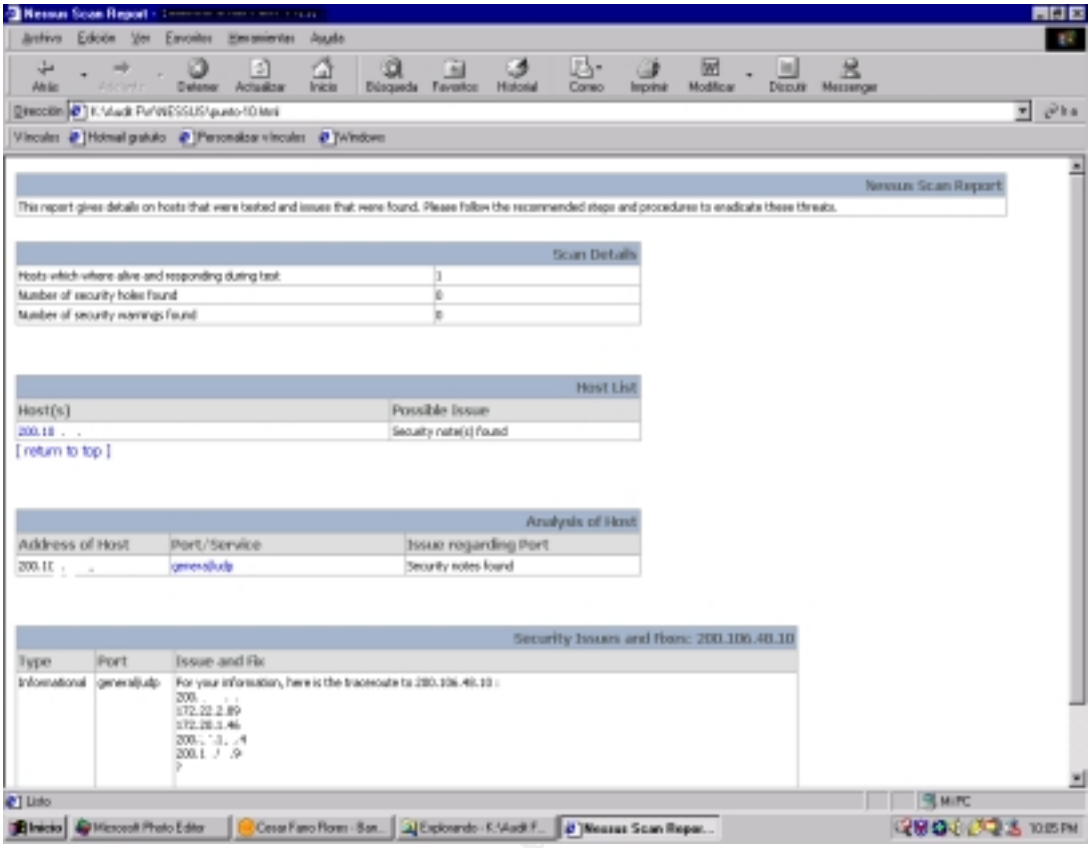
4.1.14. Additional comments

The scope of this tests were to find vulnerabilities in the firewall environment, so we haven't do any testing for the application, neither define some policies for the control of the internal network.

4.1.15. Testing the services

The Security Policy consists in ensure that any security device never responds what version or application it has installed. These information could be used by hackers in order to enter into our system.

For PIX itself, Nessus will give us the following warning message (in case it will find a hole of security). It does not detect vender and software version, because the hacker could use known PIX vulnerability to launch various attacks against GIAC.



Nessus Scan Report

This report gives details on hosts that were tested and issues that were found. Please follow the recommended steps and procedures to eradicate these threats.

Scan Details	
Hosts which where alive and responding during test	1
Number of security holes found	0
Number of security warnings found	0

Host List	
Host(s)	Possible Issue
200.10 . .	Security note(s) found

[return to top]

Analysis of Host		
Address of Host	Port/Service	Issue regarding Port
200.10 . .	general/udp	Security notes found

Security Issues and Fixes: 200.106.40.10		
Type	Port	Issue and Fix
Informational	general/udp	For your information, here is the traceroute to 200.106.40.10 : 200. . . . 172.22.2.89 172.28.1.46 200.1. .1. .4 200.1 . / .9 ?

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

4.1.16. Final conclusions and recommendations

According to the developed tests, the firewall responds correctly.

- The firewall does not respond to OS fingerprint detection.
- The firewall responds correctly to non-standard IP packets issues.
- Only the configured open ports are accessible from Internet.
- The filters of the perimeter router gives an extra protection. The recommendation is not permit traffic coming from an "reserved IP" like RFC1918 and IANA reserved range of IP. This protects against IP spoofing
- Enforce the IDS Policies in order to detect Port Scan Attacks more efficiently.
- Develop an strategy to get a bogus version information for public applications that are running in the servers. This strategy must be implemented not only for external attacks but also internal attacks.
- It is very important get some level of security in the different switches around the network because you don't know if an internal hacker might plug his/her laptop into empty jet port and start to cause damage. We are thinking in develop and strategy based in 802.1x so that the internal users at the moment to start the communications in the network can authentication in order to get the resources in a right way.
- It is strongly recommended to implement content analysis server to scan a malicious code , worms ; mail content , malicious code java. Also install a new server URL – Filtering .
- Enforce the policies in the internal firewalls to conducted a right way according the security policies of the GIAC Enterprise.

5. DESIGN UNDER FIRE

The following GIAC architecture was developed by Mr. Terry Hasford published March 2003,

(http://www.giac.org/practical/GCFW/Terry_Hasford.pdf)

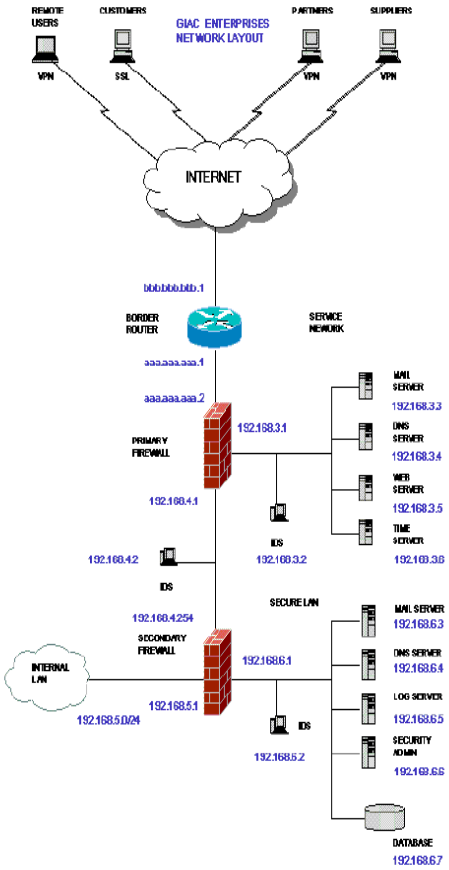


Figure 1 - GIAC Enterprises Network

Mr Terry Hasford chose CISCO PIX 515E "Unrestricted " software license (515E-UR) . The CISCO PIX 515E (UR) model uses OS **Release 6.2(1)** and PIX Device Manager **2.0 (1),** hardware based VPN Accelerator with 168 bit Triple DES IPSec which has a VPN throughput of 63 Mbps.

In his security design he used Sun One Web Server on the Solaris 9 Operating System which includes **a SunScreen 3.2** Firewall for host-based protection and BIND Version 9.2.1 as DNS Server.

Following , I'll attempt to run few attacks against this network to test it´s robustness.

5.1.1. Firewall Attack

Identify the Vulnerability

Several very useful websites of information that can obtained known vulnerabilities. We start our research but doing a search through the Bugtraq database on the securityFocus website (http://www.securityfocus.com) . Our search for CISCO PIX Firewall version 6.2(1) revealed the following vulnerabilities :

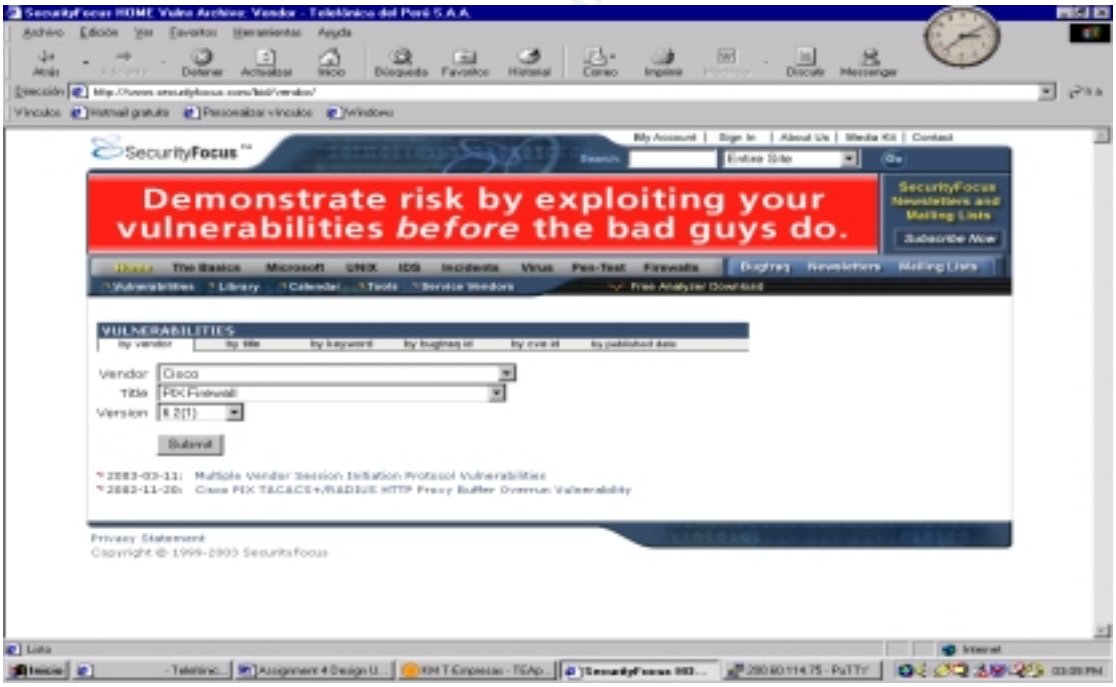


Figure - 1

A total of two were found :

1. Multiple Vendor Session Initiation Protocol Vulnerabilities .
2. Cisco PIX TACACS+/RADIUS HTTP Proxy Buffer Overrun Vulnerability

The second vulnerability does not affect this firewall because in the Terry Hasford Design , he does not use any external device Radius, TACAS+ Server to AAA : Authentication, Authorization and Accounting.

5.1.1.1. Description of the First Vulnerability

To exploit the vulnerability we need to gather additional information regarding the details of this particular vulnerability. A detailed description of this vulnerability can be viewed at :

http://www.cisco.com/warp/public/707/cisco-sa-20030221-protos.shtml#summary

and at http://www.ee.oulu.fi/research/ouspg/protos/testing/c07/sip/

SIP is the Internet Engineering Task Force (IETF) standard for multimedia conferencing over IP. SIP is an ASCII-based, application-layer control protocol (defined in RFCs 2543 and 3261) that can be used to establish, maintain, and terminate calls between two or more endpoints. SIP is an application-layer control protocol that can establish, modify, and terminate multimedia sessions (conferences) such as Internet telephony calls.

SIP is a text-based protocol and uses the UTF-8 charset. **A SIP message is either a request from a client to a server, or a response from a server to a client**. Session Description Protocol (SDP) for describing multimedia sessions." - RFC3261

The vulnerabilities identified can be easily and repeatedly demonstrated with the use of the **OUSPG "PROTOS" Test Suite for SIP**. This suite is designed to test the design limits of the implementation of the SIP protocol, **specifically the SIP INVITE messages that are used in the initial call setup between two SIP endpoints.**

The Cisco PIX Firewall **may reset when receiving fragmented SIP INVITE messages.** As the **SIP fixup does not support fragmented SIP messages**, this has been resolved to now drop SIP fragments. This vulnerability is documented as Cisco Bug ID CSCdx47789.

5.1.1.2. Design an Attack based on the vulnerability

We know that Terry Hasford has fixup to "SIP" enable, because the default configuration of CISCO PIX Firewall has the following fixup configured :

fixup protocol ftp 21
fixup protocol http 80
fixup protocol h323 h225 1720
fixup protocol h323 ras 1718-1719
fixup protocol rsh 514
fixup protocol rtsp 554
fixup protocol smtp 25
fixup protocol sqlnet 1521
fixup protocol sip 5060
fixup protocol skinny 2000

Table 1

So We have design of the attack by using an exploit to execute the attack.

The exploit was found in : http://www.ee.oulu.fi/research/ouspg/protos/testing/c07/sip/

The exploit is only intended for demonstration purposes and is harmless as it is.

Simplest of them only executes some harmless commands in the target system, typically with the privileges of the vulnerable process. Some only provide a demonstration by causing a Denial of Service (DoS) against the software.

To support the vulnerability reports to the respective vendors, following exploits were developed:

- Buffer overflow exploit allowing execution of arbitrary code was demonstrated against one terminal product and one proxy product running on a general purpose operating system.

- Denial of service was demonstrated against the remaining products identified as vulnerable.

Fortunately for us, a exploit for the crafting these packets has been already been built by http://www.ee.oulu.fi/research/ouspg/protos/testing/c07/sip/, we were able to get a copy of **"c07-sip-r1.jar"** . This is a command line utility written in Java, we have running it over solaris.

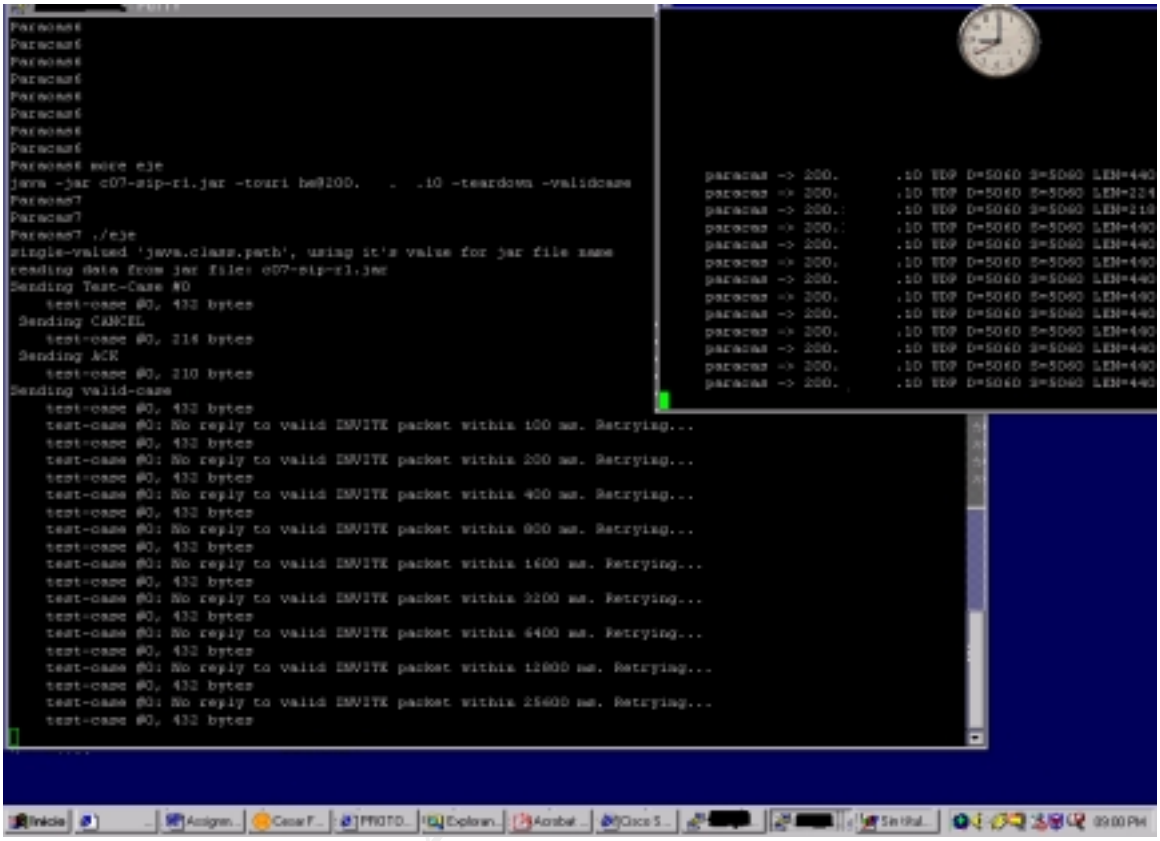


Figure - 2

5.1.1.3. Explain results :

In Normal Conditions this attack would be successfully considering Terry Hasford Design has a IP Phones ,Call Managers and so on.

5.1.1.4. Suggest Countermeasures :

Based in the analysis of this vulnerability Cisco has fixed this problem by making the update the firewall to 6.2(2) version. Is very important to get information about vulnerabilities in the vendor's website and specialized groups of security in order to maintain this process.

5.1.2. Distributed Denial of Service to the GIAC Web Server

In this test we are using a Tribe FloodNet 2k to can execute it is type of attack, the program can download from : http://packetstormsecurity.nl/distributed/tfn2k.tgz

In the following link, there are an Analysis of the TFN2.

http://packetstormsecurity.nl/distributed/TFN2k_Analysis-1.3.txt

TFN2K allows masters to exploit the resources of a number of agents in order to coordinate an attack against one or more designated targets. Currently, UNIX, Solaris, and Windows NT platforms that are connected to the Internet, directly or indirectly, are susceptible to this attack. However, the tool could easily be ported to additional platforms.

TFN2K is a two-component system: a command driven client on the master and a daemon process operating on an agent. The master instructs its agents to attack a list of designated targets. The agents respond by flooding the targets with a barrage of packets.

Multiple agents, coordinated by the master, can work in tandem during this attack to disrupt access to the target. Master-to-agent communications are encrypted, and may be intermixed with any number of decoy packets. Both master-to-agent communications and the attacks themselves can be sent via randomized TCP, UDP, and ICMP packets. Additionally, the master can falsify its IP address (spoof). These facts significantly complicate development of effective and efficient countermeasures for TFN2K.

Its important can match the MD5 code of each source where we downloaded the program.

MD5SUMS

28c9ca45a0efc86aa4ce79ea04f8a481	Makefile
7d45db74140a457966d1b6e5abd15b53	src/Makefile
be00356daefa5dc90e7838acdf24f898	src/aes.c
640aeacbd88ee76789e980bcff48642f	src/aes.h
4a963f419f2e47f5279c38faf05c39b1	src/base64.c
8f6ab658ecc6985432931995d797b52a	src/cast.c
57799312d11c174f3089dd2165a51104	src/config.h
7addb56200ebd7f8d438a15b5ccf85b8	src/disc.c
d7f4138165a5a13981f36c7a6804d9e5	src/flood.c
12e38b0e674de1b763ecac60b3fd6366	src/ip.c
83b151072d26250cf608e81105c3bd01	src/ip.h
1786c88475b5188340240539813e5d1f	src/mkpass.c

38cac21f5ba17909ea251d182da9f1a9	src/process.c
4b502ea1b820b0f9b210b8eae01afc2b	src/td.c
4341813bcce5e5caf9de53d8f2749d4c	src/tfn.c
93461e1f5016be38a15f674bf92e0dc8	src/tribe.c
562f6979a23e4a8c9852ee11b7d1f379	src/tribe.h

Now when the program is installed then we have to compromised 50 Cable/DSL Modem to can installed the Agents. The commands available can be viewed by typing "/tfn".

The results are :

[-P protocol] Protocol for server communication. Can be ICMP, UDP or TCP Uses a random protocol as default

[-D n] Send out n bogus requests for each real one to decoy targets [-S host/ip] Specify your source IP. Randomly spoofed by default, you need

to use your real IP if you are behind spoof-filtering routers

[-f hostlist] Filename containing a list of hosts with TFN servers to contact

[-h hostname] To contact only a single host running a TFN server

[-i target string] Contains options/targets separated by '@', see below

[-p port] A TCP destination port can be specified for SYN floods

<-c command ID>

0 - Halt all current floods on server(s) immediately

1 - Change IP antispoof-level (evade rfc2267 filtering) usage: -i 0 (fully spoofed) to -i 3 (/24 host bytes spoofed)

2 - Change Packet size, usage: -i <packet size in bytes>

3 - Bind root shell to a port, usage: -i <remote port>

4 - UDP flood, usage: -i victim@victim2@victim3@...

5 - TCP/SYN flood, usage: -i victim@... [-p destination port]

6 - ICMP/PING flood, usage: -i victim@...

7 - ICMP/SMURF flood, usage: -i victim@broadcast@broadcast2@...

8 - MIX flood (UDP/TCP/ICMP interchanged), usage: -i victim@...

9 - TARGA3 flood (IP stack penetration), usage: -i victim@...

10 - Blindly execute remote shell command, usage -i command

For this attack ,We will use the following command, to the web server :

bash-2.05$ ./tfn -f listserver.txt -p 80 -c 5 aaaa.aaa.aaa.3,

This command will cause the list of server send an stream of SYN packets to port 80 on the GIAC Web Server at aaaa.aaa.aaa.3, in this attack TFN2K will use a differents IP Source address spoofed .

5.1.2.1. Countermeasures for the DDOS Attack

Distributed Denial of Service attacks are very difficult to protect because it attack is coming from multiple valid source address installed on agents. Some devices such as routers, firewalls, can detect the SYN Attacks . Also if the Primary Firewall starts to reject/drop SYN Requests because it is configured to prevent SYN Attacks , it also can reject/drop legitimate traffic and generated a DOS to legitimate clients. There are no absolute means to stop TFN2K DOS Attack , here are some steps that can be used to minimize the affects of an attack :

- Configure on the firewalls capabilities to detect and block SYN floods.
- Use anti-spoofing rules on borders routers and firewalls.
- Use Bandwitdh management tools to know when the traffic is abnormal, also use the log in your router, firewall to identify the source IP generating the attack.
- Block all ICMP,UDP,TCP,RPC traffic that is not required.
- Increase the memory allocated for established connections . This will take up more memory on the server but it may allow some legitimate traffic to get through.
- Is very important know how work your Network about the traffic , services, protocols , potential range of clients which permit when the traffic is high , what type of service is more useful , what service is the most width to the firewalls, routers, switches, servers. To identify a distributed denial of service.

5.1.3. Attack against to the GIAC Web Server

5.1.3.1. Select an Attack and explain de reasons for choosing that target

The web server form GIAC Enterprise is Sun One Web Server on the Solaris 9 Operating System which includes a SunScreen 3.2.

We have chose an internal web server which runs important information to the clients and also its server be communicating with the Data Base Server.This server contains important and sensitive information for the enterprise.

5.1.3.2. Describe the process to compromise the target

We have used a type of attack called "side-channel attack". This attack employs unusual methods (unusual being in the eye of the beholder) that have little to do with the security concepts underlying a system.

In this case we are focusing in an implementation of SSL that, through analysis of the timing of certains operations, can reveal us sensitive information.
This information is enough for an adaptive attack that ultimately obtain plaintext of a target block of ciphertext.

OpenSSL CBC Error Information Leakage Weakness

http://www.securityfocus.com/bid/6884/discussion/

The information loss was reduced in OpenSSL versions 0.9.6i and 0.9.7a. It is not known if other implementations are vulnerable to this or similar weaknesses.

*It should be noted that this attack is reportedly difficult to exploit and requires that the adversary be a man-in-the-middle.

The following exploit was provided by **Martin Vuagnox** :

- /data/vulnerabilities/exploits/omen-1.1.tar.gz

bash-2.05$omen -l 993 -r aaa.aaa.aaa.3:80 -a 0

5.1.3.3. Suggest Countermeasures

Given the complexity of today's computer systems Windows, Linux, UNIX – opportunities for doing damage through these channels are plentiful.
One way to prevent these kinds of attacks is to examine all of the sources of information at every nook and cranny of an application. Thats because is very important to manage a Log management Central.

5.1.4. Recommendations :

This network could be vulnerable to these attack if GIAC Enterprise not manage a specific Log Management Central.

In summary, each of the three types of attacks and exploit could happen today. The architecture that I chose shows a good foundation for layered security which mitigates these risks.

It is also recommendation that GIAC has a group dedicated to Incidents prepared for these attacks and events.

6. REFERENCES

Spitzner, Lance. "Auditing Your Firewall Setup". 12 December 2000. URL: http://www.spitzner.net/audit.html (January 2003).

Fyodor. "Nmap network security scanner man page". 2003. URL: http://www.insecure.org/nmap/data/nmap_manpage.html (January 2003).

Fyodor. "The Art of Port Scanning". 6 September 1997. URL: http://www.insecure.org/nmap/nmap_doc.html (January 2003).

Mixter. "Tribe FloodNet 2k edition, Distributed Denial Of Service Network". URL: http://mixter.void.ru/tfn2k.tgz (February 2003).

Barlow, Jason. "Axent releases a full TFN2K Analysis". 8 March 2000. URL: http://www.securiteam.com/securitynews/5YP0G000FS.html (February 2003).

The SANS Institute. Firewalls 101: Perimeter Protection with Firewalls, Track 2.2. Bethesda: SANS Press, 2002.

The SANS Institute. Firewalls 102: Perimeter Protection and Defense In-Depth, Track 2.3. Bethesda: SANS Press, 2002.

The SANS Institute. VPNs and Remote Access, Track 2.4. Bethesda: SANS Press, 2002.

Farrell, James. **IP Fragmentation Attacks on Checkpoint Firewalls.** April 2001. (http://www.sans.org/rr/firewall/frag_attacks.php)

Common Vulnerabilities and Exposures
http://www.cve.mitre.org

CERT Coordination Center
www.cert.org

Intrusion Detection System and Netfilter Firewall

www.snort.org / www.netfilter.org

VPN Concentrator

http://www.cisco.com/en/US/products/hw/vpndevc/ps2284/prod_configuration_examples_list.html

Cisco. Cisco PIX Firewall and VPN Configuration Guide, Version 6.2. URL: http://www.cisco.com/univercd/cc/td/doc/product/iaabu/pix/pix_sw/v_62/config/index.htm

7. APENDICE

7.1. Technical Analisys of the Tribe Flood Network 2000

-=

http://packetstormsecurity.nl/distributed/TFN2k_Analysis-1.3.txt

TFN2K - An Analysis
Jason Barlow and Woody Thrower
AXENT Security Team
February 10, 2000 (Updated March 7, 2000)
Revision: 1.3

Abstract

This document is a technical analysis of the Tribe Flood Network 2000 (TFN2K) distributed denial-of-service (DDoS) attack tool, the successor to the original TFN Trojan by Mixter. Additionally, countermeasures for this attack are also covered. This document assumes a basic understanding of DDoS attacks. Analyses of related DDoS attack tools such as Stacheldraht and Trinoo are not presented here. For information about DDoS attacks and TFN2K's cousins, please refer to the following documents:

http://www2.axent.com/swat/News/ddos-explanation.htm
http://staff.washington.edu/dittrich/misc/trinoo.analysis
http://staff.washington.edu/dittrich/misc/tfn.analysis
http://staff.washington.edu/dittrich/misc/stacheldraht.analysis
http://packetstorm.securify.com/distributed
http://www.cert.org/advisories/CA-2000-01.html
http://www.cert.org/advisories/CA-99-17-denial-of-service-tools.html
http://www.cert.org/advisories/CA-98-13-tcp-denial-of-service.html
http://www.cert.org/incident_notes/IN-99-07.html
http://www.sans.org/y2k/solaris.htm
http://www.fbi.gov/nipc/trinoo.htm
http://www.fbi.gov/pressrm/pressrel/pressrel99/prtrinoo.htm

Terminology

The terminology used in DDoS analyses is often confusing. For clarity, we use the following:

Client - an application that can be used to initiate attacks by sending commands to other components (see below).

Daemon - a process running on an agent (see below), responsible for receiving and carrying out commands issued by a client.

Master - a host running a client

Agent - a host running a daemon

Target - the victim (a host or network) of a distributed attack

Overview - What is TFN2K?

TFN2K allows masters to exploit the resources of a number of agents in order to coordinate an attack against one or more designated targets. Currently, UNIX, Solaris, and Windows NT platforms that are connected to the Internet, directly or indirectly, are susceptible to this attack. However, the tool could easily be ported to additional platforms.

TFN2K is a two-component system: a command driven client on the master and a daemon process operating on an agent. The master instructs its agents to attack a list of designated targets. The agents respond by flooding the targets with a barrage of packets. Multiple agents, coordinated by the master, can work in tandem during this attack to disrupt access to the target. Master-to-agent communications are encrypted, and may be intermixed with any number of decoy packets. Both master-to-agent communications and the attacks themselves can be sent via randomized TCP, UDP, and ICMP packets. Additionally, the master can falsify its IP address (spoof). These facts significantly complicate development of effective and efficient countermeasures for TFN2K.

TFN2K - The Facts

* Commands are sent from the master to the agent via TCP, UDP, ICMP, or all three at random.

  Targets may be attacked with a TCP/SYN, UDP, ICMP/PING, or BROADCAST PING (SMURF) packet flood. The daemon may also be instructed to randomly alternate between all four styles of attack.

* Packet headers between master and agent are randomized, with the exception of ICMP, which always uses a type code of ICMP_ECHOREPLY (ping response). Unlike its predecessors, the TFN2K daemon is

completely silent; it does not acknowledge the commands it receives. Instead, the client issues each command 20 times, relying on probability that the daemon will receive at least one. The command packets may be interspersed with any number of decoy packets sent to random IP addresses.

* TFN2K commands are not string-based (as they are in TFN and Stacheldraht). Instead, commands are of the form "+<id>+<data>" where <id> is a single byte denoting a particular command and <data> represents the command's parameters. All commands are encrypted using a key-based CAST-256 algorithm (RFC 2612). The key is defined at compile time and is used as a password when running the TFN2K client.

* All encrypted data is Base 64 encoded before it is sent. This holds some significance, as the payload should be comprised entirely of ASCII printable characters. The TFN2K daemon uses this fact as a sanity-test when decrypting incoming packets.

* The daemon spawns a child for each attack against a target. The TFN2K daemon attempts to disguise itself by altering the contents of argv[0], thereby changing the process name on some platforms. The falsified process names are defined at compile time and may vary from one installation to the next. This allows TFN2K to masquerade as a normal process on the agent. Consequently, the daemon (and its children) may not be readily visible by simple inspection of the process list. All packets originating from either client or daemon can be (and are, by default) spoofed.

* The UDP packet length (as it appears in the UDP header) is three bytes longer than the actual length of the packet.

* The TCP header length (as it appears in the TCP header) is always zero. In legitimate TCP packets, this value should never be zero.

* The UDP and TCP checksums do not include the 12-byte pseudo-header, and are consequently incorrect in all TFN2K UDP and TCP packets.

Detecting TFN2K - The Signature

All control communications are unidirectional, making TFN2K extremely problematic to detect by active means. Because it uses TCP, UDP, and ICMP packets that are randomized and encrypted, packet filtering and other passive countermeasures become impractical and inefficient. Decoy packets also complicate attempts to track down other agents participating in the denial-of-service network.

Fortunately, there are weaknesses. In what appears to be an oversight (or a bug), the Base 64 encoding (which occurs after encryption) leaves a telltale fingerprint at the end of every TFN2K packet (independent of protocol and encryption algorithm). We suspect it was the intent of the author to create variability in the length of each packet by padding with one to sixteen zeroes. Base 64 encoding of the data translates this sequence of trailing zeros into a sequence of 0x41's ('A'). The actual count of 0x41's appearing at the end of the packet will vary, but there will always be at least one. The padding algorithm is somewhat obscure (but predictable) and beyond the scope of this document. However, the presence of this fingerprint has been validated both in theory and through empirical data gathered by dumping an assortment of command packets.

A simple scan for the files tfn (the client) and td (the daemon) may also reveal the presence of TFN2K. However, these files are likely to be renamed when appearing in the wild. In addition to this, both client and daemon contain a number of strings that can be found using virus scanning methods. Below is a partial list of some of the strings (or sub-strings) appearing in TFN2K:

NOTE: Scanners should look for pattern combinations unlikely to appear in legitimate software.

TFN2K Client (tfn)

[1;34musage: %s <options>
[-P protocol]
[-S host/ip]
[-f hostlist]
[-h hostname]
[-i target string]

[-p port]
<-c command ID>
change spoof level to %d
change packet size to %d bytes
bind shell(s) to port %d
commence udp flood
commence syn flood, port: %s
commence icmp echo flood
commence icmp broadcast (smurf) flood
commence mix flood
commence targa3 attack
execute remote command

TFN2K Daemon (td)

tribe_cmd *
tfn-daemon **
tfn-child **

* Mixter wisely avoids embedding clear-text strings in the TFN2K daemon. However, tribe_cmd, the one function unique to the daemon, is clearly visible and can be detected with any standard grep utility.

** Because, this text is likely to be modified in many TFN2K installations, it may be problematic to definitively identify a TFN2K daemon by traditional virus-scanning means.

TFN2K Daemon and Client (tfn and td)

security_through_obscurity *
D4 40 FB 30 0B FF A0 9F **
64 64 64 64 ... ***
ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/
/dev/urandom
/dev/random
%d.%d.%d.%d
sh ****
ksh ****

command.exe *****
cmd.exe *****

* This is a function whose definition is generated at compile time. This
is a strong (and probably unique) signature.

** This byte pattern is present in both client and daemon, and
represents the first eight bytes in the CAST-256 encryption table
(displayed in little-endian byte ordering here).

*** A contiguous 128-byte sequence of 0x64 values reveals the presence of
the static table used in the Base 64 decoding algorithm.

**** Unix and Solaris systems only

***** Windows NT systems only

The TFN2K binaries may be stripped of clear-text method and variable names, making it difficult to definitively identify the daemon by conventional string-based scanners.

Defeating TFN2K - A Strategy

There is no known way to defend against TFN2K denial-of-service attacks. The most effective countermeasure is to prevent your own network resources from being used as clients or agents.

Prevention

* Configure your router to do egress filtering, preventing spoofed traffic from exiting your network. Refer to http://www.sans.org/y2k/egress.htm for more information.

* Ask your ISP to configure their router to do ingress filtering on your network, preventing spoofed traffic reaching the Internet from your network. Refer them to RFC 2267.

* Use a firewall that exclusively employs application proxies. This should effectively block all TFN2K traffic. Exclusive use of application proxies is often impractical, in which case the allowed non-proxy services should be kept to a minimum.

* Disallow unnecessary ICMP, TCP, and UDP traffic. Typically only ICMP type 3 (destination unreachable) packets should be allowed.

* If ICMP cannot be blocked, disallow unsolicited (or all) ICMP_ECHOREPLY packets.

* Disallow UDP and TCP, except on a specific list of ports.

* Spoofing can be limited by configuring the firewall to disallow any outgoing packet whose source address does not reside on the protected network.

* Take measures to ensure that your systems are not vulnerable to attacks that would allow intruders to install TFN2K.

Detection

* Scan for the client/daemon files by name.

* Scan all executable files on a host system for patterns described in the previous section.

* Scan the process list for the presence of daemon processes.

* Examine incoming traffic for unsolicited ICMP_ECHOREPLY packets containing sequences of 0x41 in their trailing bytes. Additionally, verify that all other payload bytes are ASCII printable characters in the range of (2B, 2F-39, 0x41-0x5A, or 0x61-0x7A).

* Watch for a series of packets (possibly a mix of TCP, UDP, and ICMP) with identical payloads.

Response

Once TFN2K has been identified on a host system, it is imperative that the authorities be notified immediately so that the perpetrators can be traced. Because a TFN2K daemon does not acknowledge the commands it receives, it is likely the client will continue to transmit packets to the agent system. Additionally, a hacker observing the absence of flood activity, may attempt to reestablish direct contact with the agent system to determine the nature of the problem. In either case, the communication can be traced.

TFN2K is traceable but requires a timely response on the part of the victim. If you believe you have been the victim of TFN2K or any other DDoS attack, please contact your local authorities. In the United States, contact your local FBI office. FBI contact information can be obtained from:

http://www.fbi.gov/contact/fo/fo.htm

Summary

TFN2K and other DDoS attack signatures are under continuous investigation by AXENT Technologies. As more information becomes available, this document will be updated.

Contact Information

If you have questions or comments regarding this article or other security developments, send e-mail to securityteam@axent.com.