SANS GIAC Firewalls, Perimeter Protection and VPNs

Practical Assignment

Parliament Square SANS Assignment

Submitted by:
Orazio Mistretta
5-Oct-2001

1 Assignment 1 – Security Architecture

GIAC Enterprises is a growing Internet Startup that expects to earn $200 million per year in the sales of online fortune cookie sayings. GIAC has just completed a merger/acquisition and so has a need to interact with customers, suppliers and partners.

Since GIAC is a startup, it provides a clean state where proper security policies and practices can be implemented right from the start. In defining a Security Architecture, GIAC has to develop a Security Policy, a document that that includes the following steps:

1. Identify what we are trying to protect
2. Identify the threats
3. Identify how probable the threats are
4. Implement measures to protect GIAC assets in a cost-effective manner
5. Review the process continuosly and make improvements each time a weakness is found

Identifying the Assets
Generally the resources that are to be protected fall in the following categories:

1. Hardware: PC, servers, printers, routers, communication lines
2. Software: Programs, applications, Operating Systems, sources
3. Data: during execution, stored on-line, archived, backups, in transit over communication media
4. People: users, administrators, partners, suppliers
5. Supplies: paper, magnetic media

Identifying the threats
The following is a list of possible threats:

1. Unauthorized access to resources
2. Unauthorized disclosure of information
3. Data corruption/system compromise
4. Denial of service
5. Natural events (floods, fires, earthquakes, etc.)

Tradeoffs
GIAC goals will be largely influenced by the following key tradeoffs:

1. Services offered versus security needs
   Each service offered to users carries its own security risks.
2. Ease of use versus security
   The easiest system to use would allow access to any user and require no password, thus providing no security at all. Obviously a condition GIAC cannot afford.
3. Cost of security
   There are many different costs to security: monetary, performance, ease of use. There are also costs associated with risks: loss of privacy, loss of data, loss of service, loss of reputation. Each type of

cost must be weighed against each type of loss.

1.1 Security Policy

As defined in RFC 2196, "a security policy is a formal statement of the rules by which people who are given access to an organization's technology and information assets must abide".

As a good starting point in definining its security policy, GIAC can use the VISA list of requirements for all e-business partners:

1. Install and maintain a working network firewall to protect data accessible via the Internet.
2. Keep security patches up-to-date.
3. Encrypt stored data accessible from the Internet.
4. Encrypt data sent across networks.
5. Use and regularly update anti-virus software.
6. Restrict access to data by business "need to know."
7. Assign unique IDs to each person with computer access to data.
8. Track access to data by unique ID.
9. Don't use vendor-supplied defaults for system passwords and other security parameters.
10. Regularly test security systems and processes

The security policy should also be based on some base principles that can be identified as follows:

1. Separation of services

   The services GIAC provides have different levels of access needs and models of trust. If possible, each service should be running on a different machine (Es. mail, DNS, WWW, DB). All unneeded services will be disabled and/or removed.

2. Deny all vs. allow all

   It is obviously impossible to provide absolute security to any connected network, but the option to turn off all services and all accesses and then selectively enable them on a case by case basis as they are needed provides more security.

3. Be a good Internet neighbor

   Certain traffic would not be allowed to leave GIAC network. GIAC systems should not be permitted to be used as DDOS agents or master, nor viruses should be propagated via infected e-mail sent by internal users.

4. Provide security at each layer

   This means that the security architecture should enforce several aspects of the security environment: physical security, OS security, TCP/IP security, application security, human security

5. Develop a security plan

   A security plan for GIAC defines the list of network services provided, which areas will provide the services, who will have access to those services, how access will be provided, who will administer the services

6. Use SSL-based web applications

   It is far more secure to provide access to data through web-enabled applications accessed via SSL secured transactions, rather than to install special software on client systems

7. Err on the side of security

Security has to be considered the most critical issue, all other factors beeing considered less relevant in case a tradeoff is performed.

8. Inform all users

   Once a security policy has been established, it should be clearly communicated to all users, including staff, management, outsourcing personnel, partners, suppliers, etc.
9. Prepare and plan for incident handling
10. Review the policy regularly

    It is mandatory to verify if the policy we adopted and the architecture we developped is successfully supporting GIAC security needs.

1.2 Implementation

Fig. 1 describes the organization of the GIAC network and its connection to the Internet.
The most relevant factors are the following:

1. Every communication to the Internet is through the firewall
2. The service network (hosting the SMTP, DNS and HTTP/HTTPS services) is isolated from the internal network and sits on a different interface card of the firewall
3. To provide VPN connectivity with merged companies and to enable roaming/home workers access we use a Cisco VPN concentrator
4. Internal hosts and servers are connected to the network through a unmanaged (no SNMP service running) switch, thus ensuring that traffic cannot be easily sniffed
5. Packet filtering on the router provides additional protection to the firewall itself
6. The DB server, the syslog server and the internal DNS server are protected by an additional internal firewall
7. The internal firewall is based on a different product than the external firewall, so in case the external perimeter is broken, the internal systems are protected by systems than are not likely to be compromised by the same vulnerabilities
8. The mail servers run an antivirus program with autodownload capability (antiviral signatures are retrieved each 8 hours from the producer ftp site).

In order to conduct business, GIAC Enterprises will require a number of network services offered to a complex user population:

Internet Customers

Domain Name Services (exposing only web and mail servers)
Internet Mail
HTTP and HTTPS access for customers and potential customers.
The web server will connect to a database server containing the fortune-cookie fortunes to be accessed by the web users. (It will be assumed the database system is listening on tcp/1521, assuming it is based on Oracle/SQL-Net.)

Suppliers

Suppliers are authors who connect to supply fortunes. They need to access the database in a limited way (to add/modify/delete their products, offers, stocks, prices).
They will use a secure web application based on SSL (https) that will query database records based on an assigned username and password.

Partners

These are international affiliates that translate and resell fortunes. No direct connection to the DB server will be provided, instead GIAC will deploy the same kind of access granted to all Internet users: through a secure SSL based web application that will query database records based on an assigned username and password.

All DB access is performed through SSL secured web access. The ability to read and/or modify data on the fortunes DB is accomplished using the authentication mechanism of the application that gives access to several reserved areas based on encrypted username/password pairings and data encryption on sensitive transactions.

In this way, even if a host on a partner/supplier network has been subverted, no direct (even if encrypted) connection to GIAC Oracle server will be granted.

The DB server is an Oracle 8i with security patches applied, while the application is based on JSP/servlets to fetch dynamic data.

Merged/acquired companies

They need a secure tunnel from the company network. Their users will be enabled access to the internal network like every roaming employee of GIAC Enterprises. In a certain fashion, a merged company can be tought as a new regional office of GIAC Enterprises, connected to the Headquarter via a VPN tunnel over Internet using a Cisco VPN concentrator.

GIAC Employees

Local and remote employees will require the same services. Remote employees should be able to do the same things that they can do with their PCs at the office. An encrypted VPN solution is therefore needed. To accomplish this goal we use the same Cisco VPN concentrator.

The services they are able to access are:

- Printers/file/directory access
- E-mail
- Web access (through a proxy)
- Authentication

GIAC Administrators

These powerful users will be enabled all the functions of ordinary users, plus the privilege of administering key elements of the GIAC network services:

- Administrative access to network devices
- DB administration
- Security administration
- Router administration

1.3 Running network services

The service network will house a web server, a mail-relay server and the external dns server. The web server will run a copy of Apache with tomcat and mod_ssl add-ons. It will be allowed communication to the internal DB server TCP port 1521.

The central mail hub for GIAC enterprises will be allowed SMTP communication to the Exchange server on the internal network, as well as SMTP service to the Internet. Inbound and outbound SMTP will be scanned for viruses by both systems.

The name resolution will be handed using a chrooted named implementation based on the latest version of bind (currently 8.2.4, see http://www.isc.org). The Internet accessible name server will resolve only the WWW and SMTP server names/addresses. The internal name server will use the Internet named as a forwarder, and will be recursive only for the systems on the internal network. No zone transfers will be allowed to the Internet, or to the internal network.

All router logs will be sent to the firewall. All firewall logs will be stored both locally and on a dedicated syslog server which resides on the internal network, but protected by an additional internal firewall that isolates it from the user community.

All traffic from the Internal (protected) network to the Service network will be prohibited, the only exception beeing SMTP between the central mail-hub and the Exchange server.

GIAC Enterprise perimeter protection will be granted by the router and the firewall configurations.

Border router functions:

- Screen for packets that have private or erroneous source addresses.
- Screen broadcasts and multicasts
- Screen for syslog packets to prevent syslog flooding
- Screen for sunrpc, ftp, and NetBIOS to keep noise generated by current worms found in the wild from filling up the firewall logs uselessly.
- Screen SNMP
- Screen for packets not destined for GIAC Enterprises Network
- Screen outbound packets from addresses other than GIAC Enterprises Public Network
- Screen outbound broadcasts and multicasts
- Drop source-routed packets
- Screen directed broadcasts
- Be hardened against unauthorized access
- Log dropped packets to the firewall syslog server

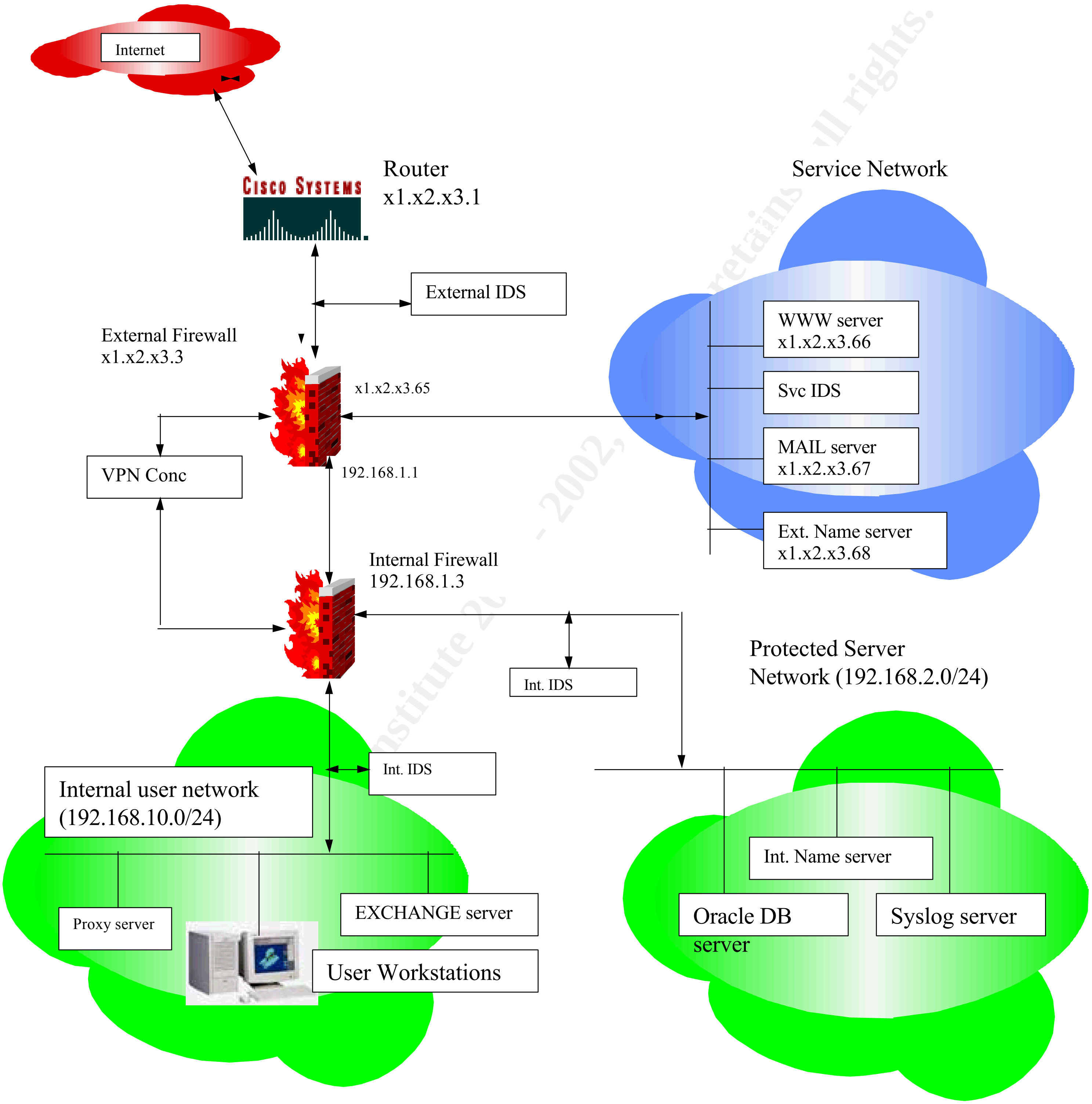
GIAC external firewall functions:

- Drop packets with IP options set
- Drop TCP with any combination that includes both SYN and FIN
- Drop TCP with no flags set (null packet)
- Drop silently packets with TTL set to 1 (prevent traceroute response)
- Screen ICMP
- Drop any packets to the firewall itself (except returned packets and ssh to the inside interface).
- Drop all packets by default
- Log all dropped packets locally and copy them to the syslog server
- Be hardened against unauthorized access

1.4 Products in use

External Router	Cisco 3660 with IOS Software 12.2
External Firewall	CheckPoint FW1 v4.1 SP5 on WNT 4.0 with SP6a
Internal Firewall	RedHat 7.1 with Ipchains
DNS servers	ISC BIND V 8.2.4
External IDS	Snort 1.8.1
Internal IDS sensors	Enterasys Networks Dragon 4.2
Svc IDS	Snort 1.8.1
Web Server	Apache 1.3.20 with Jakarta Tomcat 3.2.3 And mod_ssl 2.8.4
Mail server	Exchange server V 5.5 with SP4 AntiVirus: InoculateIT
Proxy server	Tru64 UNIX 5.1 with Squid 2.4 STABLE2
VPN concentrator	Cisco VPN Concentrator 3030 Rel 3.1

Fig. 1 – GIAC Enterprise network Security Architecture



2 Assignment 2 – Security Policy

For the purposes of the practical assignment, the public IP address range of GIAC Enterprises, Inc. will be listed as x1.x2.x3.n (where n is the host number). The strategic partner company's IP address will be listed as y1.y2.y3.n, and suppliers will be listed as z1.z2.z3.n.
The merged company public IP address will be listed as m1.m2.m3.n.
We will assume that the network connection is a single dedicated E-1 line to the Internet (2 Mbps).

We will assume the following addresses have been configured:

1. x1.x2.x3.1 Eth0 I/F of the router
2. x1.x2.x3.3 External Firewall Internet address
3. x1.x2.x3.65 Firewall svc network address
4. x1.x2.x3.129 External Firewall VPN I/F
5. x1.x2.x3.66 GIAC Public WWW server network address
6. x1.x2.x3.67 GIAC Public Mail-hub network address
7. x1.x2.x3.68 External DNS server
8. x1.x2.x3.130 VPN concentrator public address
9. 192.168.1.1 External Firewall internal network address
10. 192.168.1.3 Internal Firewall
11. 192.168.10.10 GIAC Exchange server
12. 192.168.2.11 GIAC Oracle DB server
13. 192.168.2.12 GIAC central Syslog server
14. 192.168.2.13 GIAC Internal DNS server
15. 192.168.3.0 VPN local address
16. 192.168.5.0 Merged company internal network

2.1 The external router

The external router is the first barrier against all threats coming from the Internet.
The router can make a contribution at blocking spoofing attempts, private addresses, ICMP traffic, source routing, directed broadcasts, etc.
It is therefore the first element of our perimeter defense and must be hardened as well. To connect GIAC Enterprises to Internet, we will use a Cisco 3660 router with the latest version of IOS (12.2).
Cisco published a guide on securing IP networks with Cisco routers. This guide is available at:
http://www.cisco.com/univercd/cc/td/doc/cisintwk/ics/cs003.htm
.
Another excellent paper on how to configure a screening router has been written by Frank Keeney and is available at http://pasadena.net/cisco/secure.html.

The following pointers contain documentation on how to protect against several known attacks usually found on the Internet:
http://www.cisco.com/warp/public/707/3.html (UDP diagnostic port protection)
http://www.cisco.com/warp/public/770/land-pub.shtml (Land attack protection)
http://www.cisco.com/warp/public/707/4.html (TCP SYN attacks)

http://www.pentics.net/denial-of-service/white-papers/smurf.cgi (Smurf attack)

2.1.1 Hardened router configuration

Here is reported the section of our router configuration regarding its hardening against direct attacks.

```
service password-encryption
! This removes clear-text passwords from the router configuration.
enable secret <strong-password>
! Enable secret makes the admin password a stronger hash in the
! config file
no ip source-route
! This turns off source-routed packets into the network to reduce the
! chance of spoofing.
no ip finger
no ip http server
no ip bootp server
no service tcp-small-services
no service udp-small-services
no cdp run
no snmp
! Turn off unnessesary services.
! Cisco Discovery Protocol can give away much information to a
! neighboring device.
! Small services are almost never used legitmately
!(example chargen<-> echo dos )
! Finger can give too much user info to any aggressor
! We don't want our router to be monitored, so disable snmp at all
ntp disable
! Turn off ntp on the external interface.

no ip directed-broadcast
!(on every  interface)
! Prevents directed broadcasts from gleaning information from the
! network, and it keeps the network from being a DoS amplifier

no ip unreachables
!stop icmp unreachable messages on all interfaces

logging history debugging
logging trap debugging
logging console emergencies
logging facility local7
logging x1.x2.x3.3
! We want to log to a syslog server

banner exec ^C
GIAC Enterprises
Unauthorized Access Prohibited
This System is Subject to Remote Monitoring

^C
banner motd ^C
WARNING
This system is the property of GIAC Enterprises, Inc.
Unauthorized access to this system is a violation of several Italian and international laws and is prohibited.
^C
```

```
access-list 12 permit host x1.x2.x3.3 0.0.0.255 log-input
access-list 12 deny   ip any log-input
! Log all login attempts, successful or not.
! Login is allowed only from the firewall

line vty 0 4
access-class 12 in
exec-timeout 5 0
password 7 <hash of login password here>
login
transport input telnet
! This is what prevents access to the router via access-list 12.
! Cisco IOS devices will not even respond to requests on the telnet
! port when request comes from a device denied by access-class.
```

2.1.2 Ingress and Egress filtering

By default, IOS allows all traffic. But when an ACL is added on an interface, all traffic is dropped except that which is expressely permitted. The ACL creates an implicit deny statement.
Cisco IOS supports three types of ACL:

- Standard access list
  1. Is the fastest acl (consumes less cpu cycles)
  2. Tests only the source IP address
  3. Is defined by a list number between 1 and 99
- Extended access list
  1. Tests sorce address, destination address, protocol (TCP/UDP), port
  2. Is defined by a list number between 100 and 199
  3. Can be a named access list
- Reflexive access list
  1. Is the most cpu demanding acl
  2. Is a kind of a stateful packet filter
  3. Often used as a replacement for the established command

We use named access-lists. Their benefits are that they reduce confusion and make command-line editing easier. The access list is scanned from the first rule towards the bottom, so the order of rules placement is extremely important. Inbound traffic will be screened as follows:

```
ip access-list extended Internet_Inbound
! filter inbound traffic that is from private IP addresses
deny   ip 10.0.0.0 0.255.255.255 any log
deny   ip 172.16.0.0 0.15.255.255 any log
deny   ip 192.168.0.0 0.0.255.255 any log
! filter inbound traffic that pretend to come from our addresses
deny   ip  x1.x2.x3.0 0.0.0.255 any log
! deny packets without ip address
deny   ip host 0.0.0.0 0.255.255.255 any log
! deny packet from the loopback address
deny   ip 127.0.0.0 0.255.255.255 any log
! 169.254.0.0 is used by client machines that have no DHCP lease
deny   ip 169.254.0.0 0.0.255.255 any log
! filter inbound traffic "from" broadcast or multicast nets
deny   ip 224.0.0.0 7.255.255.255 any log
deny   ip 255.0.0.0 0.255.255.255 any log
```

```
! Permit dns traffic only to our external name server
permit udp any x1.x2.x3.68 eq domain
! deny all other UDP traffic (syslog, snmp,rpc, etc.)
deny udp any any log
! Allow smtp traffic to mail hub only
permit tcp any host x1.x2.x3.67 eq smtp
!Allow http/https traffic to web server only
permit tcp any x1.x2.x3.66 eq 80
permit tcp any x1.x2.x3.66 eq 443
! Allow ESP IPSec connections to the VPN concentrator only
permit esp any x1.x2.x3.130 0.0.0.255
! Allow only established traffic into GIAC network
permit tcp any x1.x2.x3.0 0.0.0.255 gt 1023 established
! permit icmp packet too big and deny all other
permit icmp any any 3 4
deny icmp any any log
! log any traffic that was misrouted to our site
deny   ip any any log
```

Outbound traffic will be filtered as follows:

```
ip access-list extended Internet_Outbound
 permit ip x1.x2.x3.0 0.0.0.255 any
 deny   ip any any log
! permit traffic ONLY from our public address range.
```

The filters are applied as follows:

```
interface serial 0/0
 no ip directed broadcasts
 no ip proxy-arp
 no ip unreachables
 no ip redirects
 ip access-group Internet_Inbound in
 ip access-group Internet_Outbound out
```

After applying the access lists, test them with tools like nmap and/or hping or even telnet. Use netcat on a system directly connected to the router to open a range of ports (tcp 21,22,23,25,80,443) and see the traffic coming from the attacking system. Verify then the action of the filter using the following IOS command: sho ip access-list.

It should show the number of matches per rule, that is the number of packets traversing the router that were intercepted by each rule.

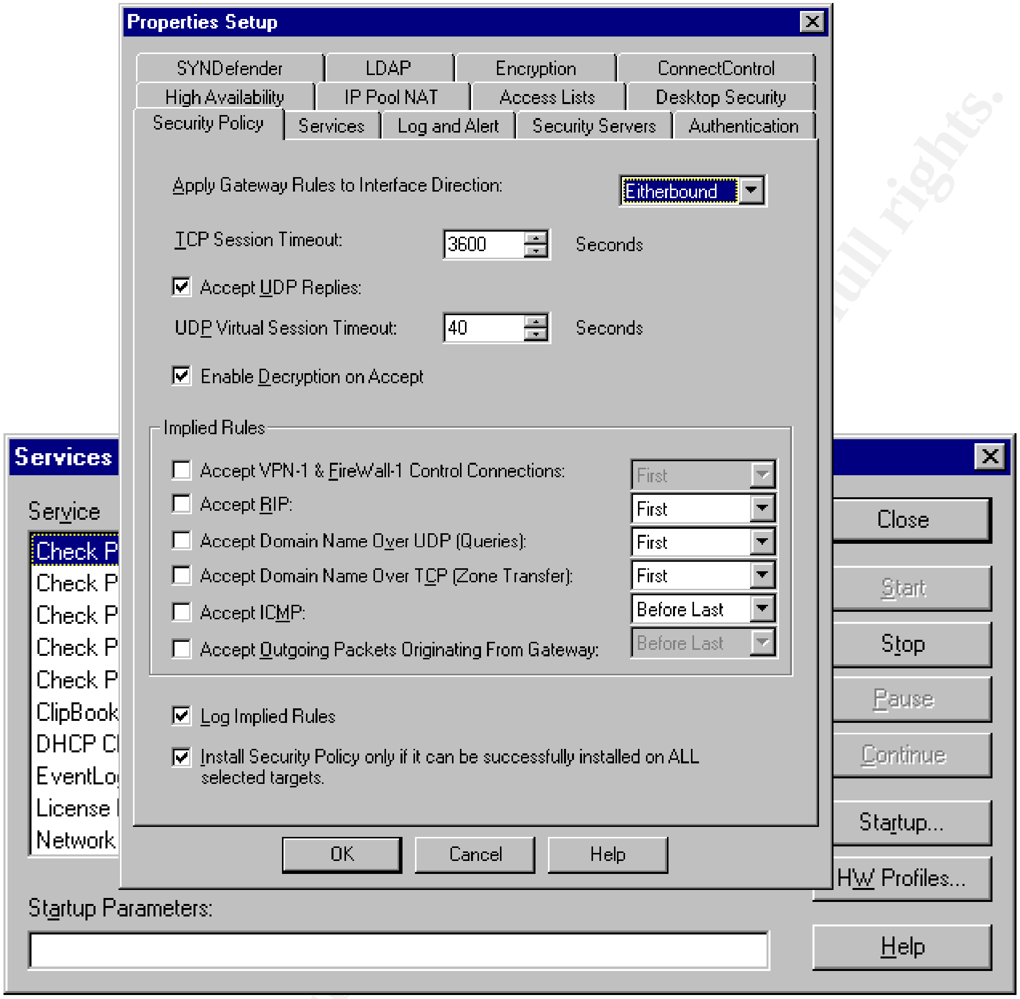
2.2 The Firewall bastion host

Our firewall is based on Checkpoint FW-1 installed on a WNT server. Checkpoint adopts a default deny policy, allowing us to block all traffic that is not explicitly permitted.

Before installing FW-1, a series of actions must be taken to harden the underlying OS, in particular:

1. Install WNT Standalone server, not as a PDC/BDC
2. Use NTFS partitions
3. Don't install IIS server
4. Don't install the software components Communications, Multimedia and Accessibility
5. Don't install IPX stack

6. Don’t configure WINS and DHCP
7. Enable IP forwarding



Apply Gateway Rules to Interface Direction: Eitherbound

TCP Session Timeout: 3600 Seconds

☑ Accept UDP Replies:

UDP Virtual Session Timeout: 40 Seconds

☑ Enable Decryption on Accept

Implied Rules

☐ Accept VPN-1 & FireWall-1 Control Connections: First

☐ Accept RIP: First

☐ Accept Domain Name Over UDP (Queries): First

☐ Accept Domain Name Over TCP (Zone Transfer): First

☐ Accept ICMP: Before Last

☐ Accept Outgoing Packets Originating From Gateway: Before Last

☑ Log Implied Rules

☑ Install Security Policy only if it can be successfully installed on ALL selected targets.

Startup Parameters:

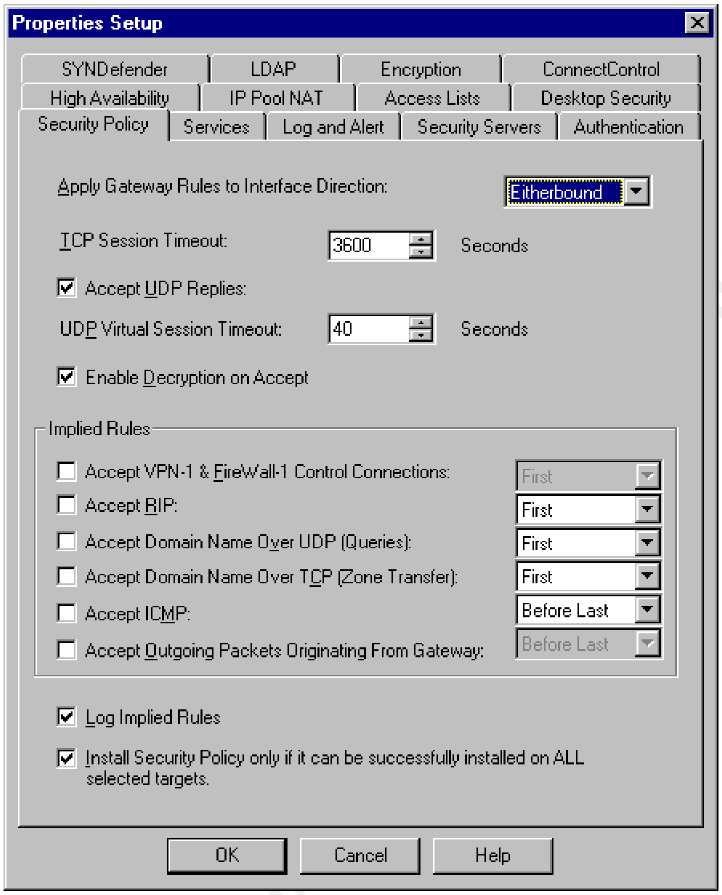
8. Use a nonexistent workgroup (not a domain!)
9. Install the latest Service Pack (SP6a at this moment)
10. Install the latest hotfixes (http://www.microsoft.com/NTServer/all/downloads.asp)
11. Remove all unneeded services (RPC Configuration, NetBIOS interface, Workstation, Server, Computer Browser)
12. Disable WINS from all the NICs
13. Disable TCP/IP NetBIOS Helper
14. Follow the NSA Windows NT Security Guidelines
15. Follow the SANS Windows NT Security: Step-by-Step 3rd ed guidelines
16. Install FW-1 and the Service Pack

Fig. 2 – WNT services

Configure the firewall following these steps:

1. Define a basic rule set policy (using the menu Policy/Properties)
2. Create objects for each node and subnet that the external firewall protects
3. Define a rule that enables its management, whilst protecting it from unwanted access

4. Define a set of rules to allow permitted traffic to traverse the firewall
5. Define NAT rules



Apply Gateway Rules to Interface Direction: Eitherbound

TCP Session Timeout: 3600 Seconds

☑ Accept UDP Replies:

UDP Virtual Session Timeout: 40 Seconds

☑ Enable Decryption on Accept

Implied Rules

☐ Accept VPN-1 & FireWall-1 Control Connections: First

☐ Accept RIP: First

☐ Accept Domain Name Over UDP (Queries): First

☐ Accept Domain Name Over TCP (Zone Transfer): First

☐ Accept ICMP: Before Last

☐ Accept Outgoing Packets Originating From Gateway: Before Last

☑ Log Implied Rules

☑ Install Security Policy only if it can be successfully installed on ALL selected targets.

Fig. 3 – Basic rule set policy

2.2.1 FW-1 complete rule set

The complete firewall-1 rule set is made by the following rules (see fig. 4):

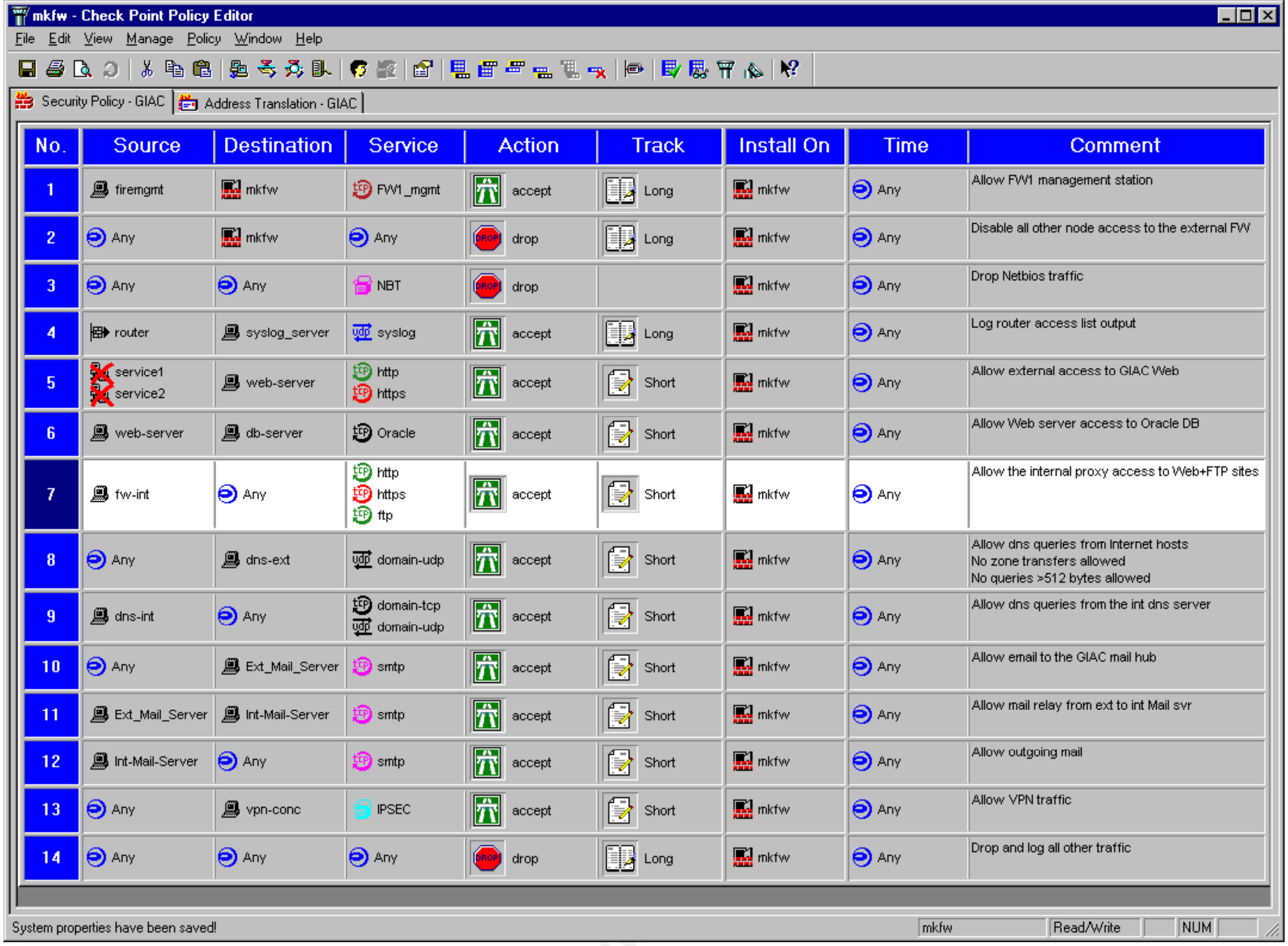
1. Allow the management station access the firewall using the service FireWall1 and log every connection
2. Deny all other accesses to the firewall from every other source
3. Deny all NBT traffic on every network interface
4. Allow the router access list output to be logged on the syslog server (TCP port 514)
5. Allow external nodes to access the Giac web server (HTTP TCP port 80, HTTPS TCP port 443)
6. Allow the Giac web server to access the fortune cookies data stored on the Oracle server (TCP port 1521)
7. Allow internal proxy access to Internet web and ftp servers (HTTP:TCP port 80, HTTPS: TCP port 443, FTP TCP port 20-21), and nat it
8. Allow external dns queries to reach the external dns server on the service net (UDP port 53)

9. Allow the internal dns server to query external servers (root servers etc.) (TCP/UDP port 53) on behalf of internal users, and nat it
10. Allow external systems to send e-mail (SMTP, TCP port 25) to the Giac central mail hub
11. Allow external mail server to relay e-mail messages to the internal mail (SMTP, TCP port 25) server where user mailboxes are defined
12. Allow the internal mail server to send e-mail to Internet servers (SMTP, TCP port 25) and nat it
13. Allow VPN connections from external sites (negotiation and IPSEC and PPTP). The protocols envolved are:
    - Authentication Header (AH), IP protocol 51
    - Encapsulation Security Payload (ESP), IP protocol 50
    - ISAKMP (Key exchange protocol), UDP port 500
    - SKIP, IP protocol 57
    - General Routing Encapsulation (GRE) for PPTP , IP protocol 47
14. Block every other packet and log it

Every packet is analyzed by the FW-1 stateful engine that scans the rule set from rule 1 downwards, until a match is found. Therefore, the order of the rules is important and care should be taken to ensure that the proper action is performed on every packet.

In addition, performance considerations dictate that the most “probable” rules should be placed in the chain before less “probable” ones, so that a more frequent packet flow descends less deeply into the chain and the relevant action be performed with less operations.

Logging of blocked packets should contain more details than expected traffic, to allow more precise determination of the intent of the source.


mkfw - Check Point Policy Editor
File
Edit
View
Manage
Policy
Window
Help
Security Policy - GIAC
Address Translation - GIAC
System properties have been saved!
mkfw
Read/Write
NUM

No.	Source	Destination	Service	Action	Track	Install On	Time	Comment
1	firemgmt	mkfw	FW1_mgmt	accept	Long	mkfw	Any	Allow FW1 management station
2	Any	mkfw	Any	drop	Long	mkfw	Any	Disable all other node access to the external FW
3	Any	Any	NBT	drop		mkfw	Any	Drop Netbios traffic
4	router	syslog_server	syslog	accept	Long	mkfw	Any	Log router access list output
5	service1 service2	web-server	http https	accept	Short	mkfw	Any	Allow external access to GIAC Web
6	web-server	db-server	Oracle	accept	Short	mkfw	Any	Allow Web server access to Oracle DB
7	fw-int	Any	http https ftp	accept	Short	mkfw	Any	Allow the internal proxy access to Web+FTP sites
8	Any	dns-ext	domain-udp	accept	Short	mkfw	Any	Allow dns queries from Internet hosts No zone transfers allowed No queries >512 bytes allowed
9	dns-int	Any	domain-tcp domain-udp	accept	Short	mkfw	Any	Allow dns queries from the int dns server
10	Any	Ext_Mail_Server	smtp	accept	Short	mkfw	Any	Allow email to the GIAC mail hub
11	Ext_Mail_Server	Int-Mail-Server	smtp	accept	Short	mkfw	Any	Allow mail relay from ext to int Mail svr
12	Int-Mail-Server	Any	smtp	accept	Short	mkfw	Any	Allow outgoing mail
13	Any	vpn-conc	IPSEC	accept	Short	mkfw	Any	Allow VPN traffic
14	Any	Any	Any	drop	Long	mkfw	Any	Drop and log all other traffic

Fig. 4 – External firewall complete rule set

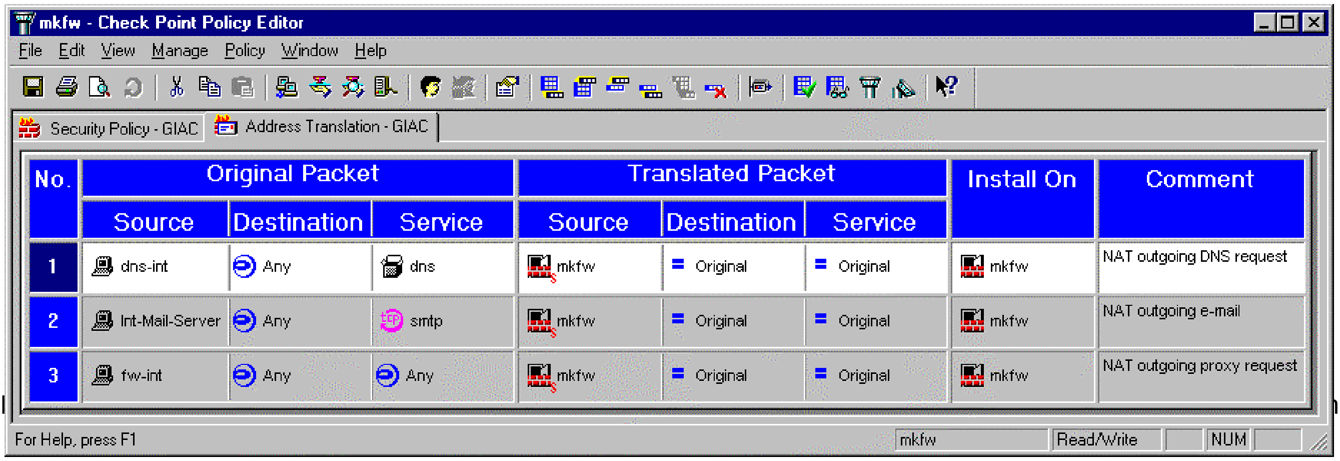
2.2.2 FW-1 Nat Configuration

We want to hide the details of our network structure to external prying eyes, so we want that the traffic originating from the internal systems leave our network with a source address of the firewall.

Giac NAT configuration is the following:

1. Internal dns requests are resolved by the internal dns server. This server is hidden by the firewall with NAT rule 1.
2. NAT rule 2 hides the address of the internal mail server, when sending e-mail outside Giac
3. NAT rule 3 hides the internal firewall that acts as a proxy allowing internal users web and ftp access

Figure 5 shows how we set up Network address Translation on the Giac firewall.



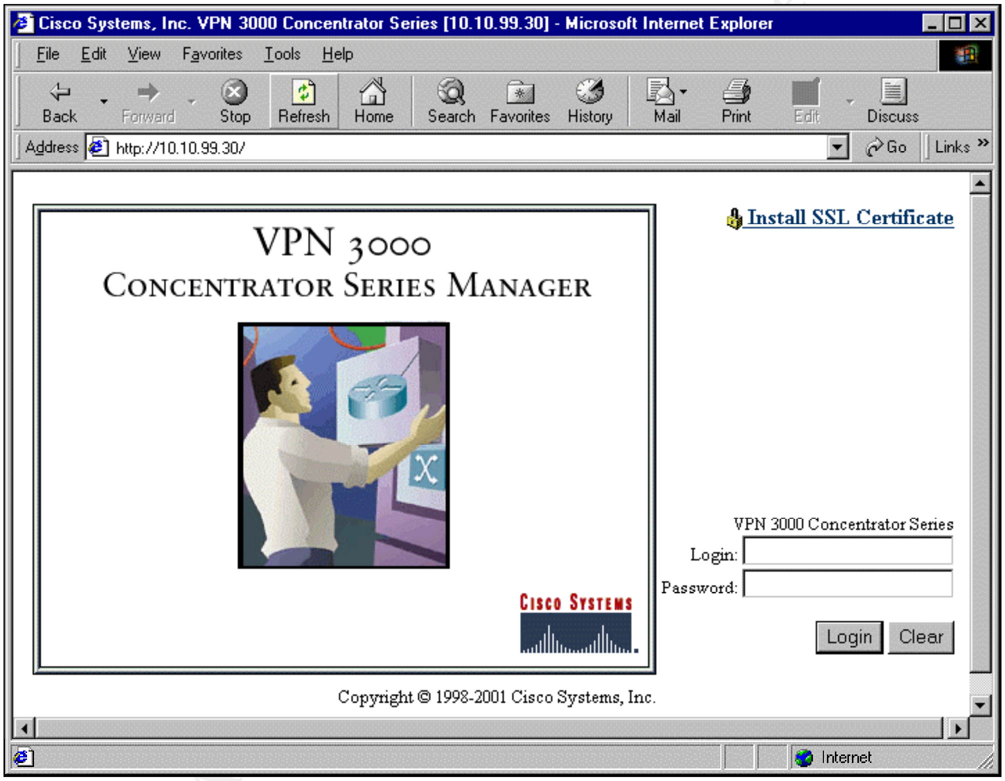
No.	Original Packet			Translated Packet			Install On	Comment
	Source	Destination	Service	Source	Destination	Service		
1	dns-int	Any	dns	mkfw	Original	Original	mkfw	NAT outgoing DNS request
2	Int-Mail-Server	Any	smtp	mkfw	Original	Original	mkfw	NAT outgoing e-mail
3	fw-int	Any	Any	mkfw	Original	Original	mkfw	NAT outgoing proxy request

Fig. 5 – NAT Configuration

2.3 VPN Configuration

A Cisco VPN Concentrator 3030 will be used to allow authorized users and/or third parties to connect to GIAC network.

We will configure the VPN Concentrator in two steps:
1. using the command line I/F we will set up the two interfaces with valid addresses/subnet masks
2. using the web interface we will configure all other protocols/parameters


Cisco Systems, Inc. VPN 3000 Concentrator Series [10.10.99.30] - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss
Address http://10.10.99.30/
Go Links
VPN 3000
Concentrator Series Manager
Install SSL Certificate
Cisco Systems
VPN 3000 Concentrator Series
Login:
Password:
Login Clear
Copyright © 1998-2001 Cisco Systems, Inc.
Internet

The VPN Concentrator Manager is an HTML-based interface that lets you configure, administer, monitor, and manage the VPN 3000 Series Concentrator with a standard web browser. To use it, you only need to connect to the VPN Concentrator using a PC and browser on the same private network as the VPN Concentrator.

Using it, the following components/parameters should be set (see http://www.cisco.com/univercd/cc/td/doc/product/vpn/vpn3000/3_1/config/index.htm):
1. Tunnelling protocol PPTP should be enabled
2. Tunnelling protocol IPSec should be enabled
3. Incoming connections should be given addresses taken from a preconfigured address pool
4. Routing to the internal network should be established using static routes

PPTP

The PPTP protocol defines mechanisms for establishing and controlling the tunnel, but uses Generic Routing Encapsulation (GRE) for data transfer.

PPTP is a client-server protocol. The VPN Concentrator always functions as a PPTP Network Server (PNS) and supports remote PC clients. The PPTP tunnel extends all the way from the PC to the VPN Concentrator.

PPTP is popular with Microsoft clients. Microsoft Dial-Up Networking (DUN) 1.2 and 1.3 under Windows 95/98 support it, as do versions of Windows NT 4.0 and Windows 2000. PPTP is typically used with Microsoft encryption (MPPE).

IPSec

IPSec provides the most complete architecture for VPN tunnels, and it is perceived as the most secure protocol. Both LAN-to-LAN connections and client-to-LAN connections can use IPSec.

In IPSec terminology, a "peer" is a remote-access client or another secure gateway. During tunnel establishment under IPSec, the two peers negotiate Security Associations that govern authentication, encryption, encapsulation, key management, etc. These negotiations involve two phases: first, to establish the tunnel (the IKE SA); and second, to govern traffic within the tunnel (the IPSec SA).

In IPSec LAN-to-LAN connections, the VPN Concentrator can function as initiator or responder. In IPSec client-to-LAN connections, the VPN Concentrator functions only as responder. Initiators propose SAs; responders accept, reject, or make counter-proposals—all in accordance with configured SA parameters. To establish a connection, both entities must agree on the SAs.

We must configure, activate, and prioritize IKE proposals before configuring LAN-to-LAN connections. We use IKE-3DES-MD5, that is we use preshared keys and MD5/HMAC-128 for authentication. Use 3DES-168 encryption.

To authenticate the peer during Phase 1 IKE negotiations, we should use a pre-shared key. This key becomes the password for the IPSec LAN-to-LAN group that is created.

Next we set up packet authentication that proves that data comes from whom we think it comes from. The IPSec ESP (Encapsulating Security Payload, IP protocol 50) protocol provides both encryption and authentication. Use ESP/MD5/HMAC-128, ESP protocol using HMAC (Hashed Message Authentication Coding) with the MD5 hash function using a 128-bit key, this is the default choice.

Next, we set up data encryption that makes the data unreadable if intercepted. We use 3DES-168, that is Triple-DES encryption with a 168-bit key. This choice is the most secure and it is the default choice.

2.4 Dns server configuration

We set up a chrooted DNS server, using a non privileged account on both the external and the internal name server. As a reference, consult the pages at http://www.dns.net/dnsrd/, http://www.cymru.com/~robt/Docs/Articles/secure-bind-template.html, http://www.pgci.ca/p_bind.html and at http://www.psionic.com/papers/dns.

This is from our startup script for named:

```
if  /usr/sbin/named -u named -g named -t /usr/local/named ; then
              echo “BIND server started ”
fi
#
```

While the config file for the externally accessible named is as follows:

```
acl internal { x1.x2.x3.0/26; x1.x2.x3.64/26 };
acl bogus {
       0.0.0.0/8;
       1.0.0.0/8;
       2.0.0.0/8;
       169.254.0.0/16;
       192.0.2.0/24;
       172.16.0.0/12;
       192.168.0.0/16;
       224.0.0.0/3;
       240.0.0.0/4;
       };

blackhole {
       bogus;};

options {
        directory "/etc/namedb";
        datasize 32M;
        coresize 4M;
        transfer-format many-answers;
        listen-on { x1.x2.x3.3; x1.x2.x3.65; };
        transfers-in 4;
        allow-recursion { x1.x2.x3.0/26; x1.x2.x3.64/26; };
        fake-iquery no;
        version "MKFW ndaemon v2.3-H";
// only our ISP name servers, secondary for our zones
// can download our zone files
       allow-transfer { XXX.XXY.1.1; XXX.XXY.1.29; };
       allow-query { internal; };
        forwarders {
                XXX.XXY.1.1;
// our ISP name servers
                ZZZ.ZZX.3.8;
        };
};

zone "giac.it" {
        type master;
        file "giac.db";
       allow-query { any; };
};
```

```
zone "x3.x2.x1.in-addr.arpa" {
        type master;
        file "giac.rev";
       allow-query { any; };
};

//
zone "0.0.127.in-addr.arpa" {
        type master;
        file "named.local";
};

//
// load the cache data last
zone "." {
        type hint;
        file "named.ca";
};
```

The internal Name server uses the following configuration :

```
acl internal { 192.168.1.0/24; };
acl bogus {
       0.0.0.0/8;
       1.0.0.0/8;
       2.0.0.0/8;
       169.254.0.0/16;
       192.0.2.0/24;
       172.16.0.0/12;
       224.0.0.0/3;
       240.0.0.0/4;
       };

blackhole {
       bogus;};

options {
        directory "/etc/namedb";
        datasize 32M;
        coresize 4M;
        transfer-format many-answers;
       listen-on { 192.168.1.1;};
        transfers-in 4;
//     recursion yes;
       allow-recursion { 192.168.1.0/24; };
       fake-iquery no;
       version "MKFW ndaemon v2.3-H";
// We have an internal secondary name server
       allow-transfer { 192.168.1.222; };
       allow-query { internal; };
        forwarders {
              x1.x2.x3.3;
        };
};

zone "giac.it" {
        type master;
        file "giac.db";
       allow-query { internal; };
};

zone "x3.x2.x1.in-addr.arpa" {
        type master;
        file "giac.rev";
        allow-query { internal; };
};

//
zone "0.0.127.in-addr.arpa" {
        type master;
        file "named.local";
};

//
// load the cache data last
zone "." {
        type hint;
        file "named.ca";
};
```

2.4.1 Proxy

We don't allow direct connection to Internet to our internal systems. In order to provide web browsing to GIAC employees, we set up a proxy using a free proxy server: Squid.

We don't allow access to porn/pedophile/gambling sites to our users. Note that during the nine to five workday the following happens:

1. 70% of all porn access happens (source: SexTracker)
2. 30 to 40% of Internet surfing is not business-related (source: IDC)
3. and more than 60% of online purchases are made (source: Nielsen//NetRatings)..

Our Squid config file has the following (not default values) setup:

```
# SQUID NETWORK OPTIONS:
# listen only on internal network
http_port 192.168.1.1:8080
#  TAG: icp_port
#          The port number where Squid sends and receives ICP queries to
#          and from neighbor caches.  Default is 3130.  To disable use
#          "0".  May be overridden with -u on the command line.
#
icp_port 0
#
# To force certain objects to never be cached.
#
acl QUERY urlpath_regex cgi-bin \?
no_cache deny QUERY
#
# To access FTP sites anonymously, use this account
ftp_user giac@
#
# ACCESS CONTROLS
# -----------------------------------------------------------------
acl all src 0.0.0.0/0.0.0.0
acl dmz src x1.x2.x3.64/255.255.255.192
acl internal src 192.168.10.0/255.255.255.0
acl manager proto cache_object
acl localhost src 127.0.0.1/255.255.255.255
acl SSL_ports port 443 563
acl Safe_ports port 80                  # http
acl Safe_ports port 21              # ftp
acl Safe_ports port 443 563         # https, snews
acl Safe_ports port 70              # gopher
acl Safe_ports port 210             # wais
acl Safe_ports port 1025-65535      # unregistered ports
acl Safe_ports port 280             # http-mgmt
acl Safe_ports port 488             # gss-http
acl Safe_ports port 591             # filemaker
acl Safe_ports port 777             # multiling http
acl CONNECT method CONNECT
http_access allow manager localhost
http_access deny manager
# Deny requests to unknown ports
http_access deny !Safe_ports
# Deny CONNECT to other than SSL ports
http_access deny CONNECT !SSL_ports
#
# Allow internal hosts access
# And finally deny all other access to this proxy
http_access deny dmz
http_access allow internal
http_access deny all
# Content filtering rules follow here
acl porn url_regex –i "/usr/local/squid/etc/porn"
acl noporn url_regex –i "/usr/local/squid/etc/noporn"
http_access deny porn !noporn
#  TAG: icp_access
#        Allowing or Denying access to the ICP port based on defined
#        access lists
#
icp_access deny all
```

Note: the files porn and noporn contain lines with a single word per line, where each word, if contained in a URL signifies that the site must be blocked acces to. We use the files reported at http://web.onda.com.br/orso/ as a starting point, and we added some more italian words to them.

Moreover, a list of candidate sites to be blocked can be downloaded from: http://www.hklc.com/squidblock/datafiles/squidblock.tgz

2.4.2 Syslog

We run the standard Tru64 UNIX syslog server at GIAC, but we protect it listing only the router, the firewalls and the internal IDS sensors in the file /etc/syslog.auth, that provides permission to send packets to the daemon. The syslog server offer only one service: syslog on TCP port 514. All other services are disabled. The server itself is thoroughly hardened, following the prescriptions of section 2.5.

On our internal syslog server, we use a modified copy of logcheck found on http://www.psionic.com/tools/logcheck-1.1.1.tar.gz to periodically scan our log files and report significant events. We need to modify the original script, because we found that on Tru64 UNIX it was capable to loose a certain amount of data when the system syslog daemon restarted. This is an event that occurs every day because we enable the authomatic directory log rotation. So we had to rescan the previous day directory each time logcheck is running, to avoid data loss (as seen on the lines containing LOGPREV).

We run the scancheck and the logcheck procedures every hour via a root crontab job.

Logcheck emails the sysadmin every denied attempt logged and every intrusion attempt detected by snort. Scancheck, on the other hand, builds up a directory containing a file per every system that produced a log. Using this procedure, we populate a directory (/usr/local/etc/scan) where each file has a name representative of an host (its IP address) that has been blocked access. In this way we have an historical record of the attempts coming from each IP address and we are able to reconstruct attacks even if they are distributed in a months timeframe

Scancheck rises an alarm window only on the internal syslog server (not on the firewall, since it has no graphic I/F) and emails both system administrators (root) a list of files in the /usr/local/etc/scan directory as provided by the command ls –lt.

Thus the list of the last IP addresses that generated an attack is more evident.

The scanlogit procedure is a simple xterm that reports an alarm whenever the events cross a limit that we set up. Of course, to enable the graphics alarm, a display manager must be running, and it is a severe security hole we don't want to open in the firewall itself, so we use this procedure only on the internal syslog server. On our syslog/alarm server we raise a limit on the number of windows opened on a screen, so we cannot be stalled by a huge attack.

Since we want to contribute to the Internet security, we will provide our summarized logs to http://www.incidents.org and/or http://dshield.org.

Here we report our scancheck procedure, as a reference to those ineterested.

```
#!/bin/sh
#
PATH=/bin:/sbin:/usr/bin:/usr/sbin:/usr/ucb:/usr/local/bin
# Person to send log activity to.
SYSADMIN=root
# Full path to logtail program.
# This program is required to run this script and comes with the package.
LOGTAIL=/usr/local/bin/scantail
ROOTD=/usr/local/etc
SCAND=/usr/local/scan
TMPDIR=/usr/local/scan/tmp
# Digital OSF/1, Irix
MAIL=Mail
HACKING_FILE=/usr/local/etc/scancheck.hacking
VIOLATIONS_IGNORE_FILE=/usr/local/etc/logcheck.violations.ignore
IGNORE_FILE=/usr/local/etc/logcheck.ignore
HOSTNAME=`hostname`
DATE=`date +%m/%d/%y:%H.%M`
umask 077
rm -f $TMPDIR/check.$$ $TMPDIR/checkoutput.$$ $TMPDIR/checkreport.$$
if [ -f $TMPDIR/check.$$ -o -f $TMPDIR/checkoutput.$$ -o -f $TMPDIR/checkreport.$$ ]; then
       echo "Log files exist in $TMPDIR directory that cannot be removed. This
may be an attempt to spoof the log checker." \
       | $MAIL -s "$HOSTNAME $DATE ACTIVE SYSTEM ATTACK!" $SYSADMIN
       exit 1
fi
# Digital OSF/1
# OSF/1 - uses rotating log directory with date & time in name
        LOGDIRS=`find /var/adm/syslog.dated/* -type d -prune -print`
        LOGDIR=`ls -dtr1 $LOGDIRS | tail -1`
       LOGPREV=`ls -dtr1 $LOGDIRS | tail -2|head -1`
       if [ ! -d "$LOGPREV" ]
        then
          echo "Can't identify previous log directory." >> $TMPDIR/checkreport.$$
        else
                $LOGTAIL  $LOGPREV/auth.log >> $TMPDIR/check.$$
                $LOGTAIL  $LOGPREV/daemon.log >> $TMPDIR/check.$$
                $LOGTAIL  $LOGPREV/kern.log >> $TMPDIR/check.$$
                $LOGTAIL  $LOGPREV/router.log >> $TMPDIR/check.$$
                $LOGTAIL  $LOGPREV/firewall.log >> $TMPDIR/check.$$
                $LOGTAIL  $LOGPREV/lpr.log >> $TMPDIR/check.$$
                $LOGTAIL  $LOGPREV/mail.log >> $TMPDIR/check.$$
                $LOGTAIL  $LOGPREV/syslog.log >> $TMPDIR/check.$$
                $LOGTAIL  $LOGPREV/user.log >> $TMPDIR/check.$$
        fi
        if [ ! -d "$LOGDIR" ]
        then
          echo "Can't identify current log directory." >> $TMPDIR/checkreport.$$
        else
                $LOGTAIL  $LOGDIR/auth.log >> $TMPDIR/check.$$
                $LOGTAIL  $LOGDIR/daemon.log >> $TMPDIR/check.$$
                $LOGTAIL  $LOGDIR/kern.log >> $TMPDIR/check.$$
                $LOGTAIL  $LOGDIR/router.log >> $TMPDIR/check.$$
                $LOGTAIL  $LOGDIR/firewall.log >> $TMPDIR/check.$$
                $LOGTAIL  $LOGDIR/lpr.log >> $TMPDIR/check.$$
                $LOGTAIL  $LOGDIR/mail.log >> $TMPDIR/check.$$
                $LOGTAIL  $LOGDIR/syslog.log >> $TMPDIR/check.$$
                $LOGTAIL  $LOGDIR/user.log >> $TMPDIR/check.$$
                $LOGTAIL  /var/log/snort/alert >> $TMPDIR/check.$$
        fi

# Set the flag variables
FOUND=0; ATTACK=0 ; HACK=0
# See if the tmp file exists and actually has data to check,
# if it doesn't we should erase it and exit as our job is done.
if [ ! -s $TMPDIR/check.$$ ]; then
       rm -f $TMPDIR/check.$$
       exit 0
fi
# Perform Searches
# Check for blatant hacking attempts
if [ -f "$HACKING_FILE" ]; then
       if $GREP -i -f $HACKING_FILE $TMPDIR/check.$$ > $TMPDIR/checkoutput.$$; then
              FOUND=1
              ATTACK=1
       fi
fi
# Do reverse grep on patterns we want to ignore
if [ -f "$IGNORE_FILE" ]; then
       if $GREP -v -f $IGNORE_FILE $TMPDIR/check.$$ > $TMPDIR/checkoutput.$$; then
              FOUND=1
       fi
fi
# Extract the lines containing the words REJECT or denied
# from the firewall and the router logs
# We also check snort logs to find the string "{TCP}" and the
# “End” of every portscan record
if [ "$ATTACK" -eq 1 ]; then
       awk '$6 ~ /REJECT:/ && $7 !~/ICMP/ {print substr($8,2,(index($8,"]")-2))}'
```

```
$TMPDIR/checkoutput.$$ > $TMPDIR/prova.$$
       awk '$12 ~ /denied/ {print substr($14,1,(index($14,"("))-1 ) }' $TMPDIR/checkoutput.$$
>> $TMPDIR/prova.$$
       awk 'NF>0 {
              for (i=1;i<=NF; i++)
              if ($i == "{TCP}")
              print substr($(i+1),1,(index($(i+1),":")-1))
              }'   $TMPDIR/checkoutput.$$ >> $TMPDIR/prova.$$
       awk '$5 ~ /End/ {print substr($9,1,(length($9)-1 )) }'  $TMPDIR/checkoutput.$$ >>
$TMPDIR/prova.$$

       sort -u $TMPDIR/prova.$$ -o $TMPDIR/prova2.$$
# Now update the files in the archive directory
       while read var
                     do
              if  [ "$var" -ne "" ]; then
                     touch $SCAND/$var
                     chmod 700 $SCAND/$var
                     grep $var $TMPDIR/checkoutput.$$ >> $SCAND/$var
                     HACK=`grep  $var $TMPDIR/checkoutput.$$| wc -l`
                     if [ "$HACK" -gt 5 ]; then
                            $ROOTD/scanlogit $var $HACK
                            HACK=0
                     fi
              fi
        done < $TMPDIR/prova2.$$
fi
# Clean Up
rm -f $TMPDIR/check.$$ $TMPDIR/checkoutput.$$ $TMPDIR/checkreport.$$
rm -f $TMPDIR/prova.$$ $TMPDIR/prova1.$$ $TMPDIR/prova2.$$
if [ "$ATTACK" -eq 1 ]; then
       ls -lt $SCAND | $MAIL -s "SECURITY: SUMMARY system probe on $DATE" $SYSADMIN
fi
```

2.4.3 Snort

We want to run an expandible IDS system on our external network, so we choose snort, a lightweight free software.

We start snort 1.8.1 using the following command:

```
     if /usr/kits/IDS/snort/snort -A fast -i ee0 -h x1.x2.x3.3/26 -l
/var/log/snort  -c /usr/kits/IDS/snort/snort.conf -D
```

We choose to start snort in Alert mode as a daemon. Snort logs its data to /var/log/snort, and updates a file named alert. The alert file is read by logcheck and scancheck to email root about snort findings and to create the archive directory under /usr/local/scan, where we store our precious attack history.

GIAC policy mandates that the Security administrator of the firewall check the snort website http://www.snort.org at least each week to fetch updates to the executable itself or to the snort signatures files.

2.5 Hardening exposed servers

The firewall and the border router constitute the perimeter defense to our network, but they cannot stop an in-band attack, where the compromise is based on the actual content of permitted traffic. To protect GIAC against such activities, we have to harden our servers, both at the operating system level and at the application layer. We follow the following guidelines:

2.5.1 Hardening Operating Systems

A general overview:
http://www.sans.org/infosecFAQ/securitybasics/host_sec.htm

The SANS/FBI top 20 Internet Security Vulnerabilities:
http://66.129.1.101/top20.htm

Securing guidelines for Windows NT/2000:
http://www.sans.org/newlook/publications/ntstep.htm
http://www.enteract.com/~lspitz/nt.html
http://www.cert.org/tech_tips/win_configuration_guidelines.html

Securing guidelines for Linux systems:
http://www.enteract.com/~lspitz/linux.html

General UNIX hardening suggestions compiled by CERT:
http://www.cert.org/tech_tips/unix_configuration_guidelines.html

2.5.2 Hardening exposed services

Web security information provided by CERT:
http://www.cert.org/other_sources/websec.html

Microsoft's Security Guides for Exchange Server:
http://www.microsoft.com/technet/security/email.asp

Microsoft's Security Guides for IIS:
http://www.microsoft.com/technet/security/web.asp

New vulnerabilities are discovered every day and patches become available to fix them. It is therefore mandatory to stay current with the latest information regarding new attack techniques, new tools available to the hacking community, new vulnerabilities found on free and on commercial products.
I include some links here for that purpose.

Searchable Securityfocus (Bugtraq) Archive:
http://www.securityfocus.com/

Searchable CERT Announcements/Bulletins:
http://www.cert.org/

Searchable Microsoft Security Bulletins:
http://www.microsoft.com/technet/security/current.asp

ISS X-force vulnerabilities and alerts database:
http://xforce.iss..net/alert

3 Assignment 3 - Audit the Security Architecture

Auditing the implementation of a security architecture is the first step that must be taken after all the installation and configuration of a site has been performed. Only a thorough audit of the perimeter defense and the internal configuration can provide the proof of the concept.

After all, GIAC is an E-commerce site, where customer info is stored, where price lists for end-users, resellers, distributors are constantly modified, where credit card info are requested for the purchase of products.

There are, therefore a series of risks that must be taken into account, and each risk has an associated cost in terms of:

1. Loss of privacy (unauthorized distribution of personal data)
2. Loss of competitivity (unauthorized distribution of sales data, catalogues, price lists, discounts, etc)
3. Loss of reputation: in case a penetration is successful customers/investors will no longer trust GIAC
4. Loss of business: in case a penetration is successful customers will no longer purchase GIAC products

Prior to beginning the audit of the GIAC Enterprise, some administrative tasks must be taken (see Kofi Arthiabah at http://www.sans.org/y2k/practical/Kofi_Arthiabah.zip):

a) Obtain a signed contract describing the extent of work to be covered. A signed contract would provide a legal framework necessary before starting any hostile activity on the customer network.
b) Obtain and review copies of the security policy
c) Identify the GIAC System Administrators, Network Admins, the Security Incidents Response Team and the Security Manager.
d) Present a project plan highlighting tasks to be performed and duration as well all the resources required to carry out the tasks.

As a reference in coducting the audit, consult the excellent book "Hacking exposed" that describes the phases and activities performed by hackers while trying to compromise/attack a network.

3.1 Phases of the auditing

3.1.1 Reconnaissance

During this phase, as much information as possible will be collected about GIAC Enterprises as possible. This will provide the initial setup for any social engineering techniques that would be subsequently applied.

Information that can be obtained about GIAC Enterprises include:

- Names, addresses and telephone numbers of executives
- Internet presence information, available from the WHOIS databases. The information gathered here

will help establish details of domain names, technical and administrative contacts, range of IP network numbers, format of internal e-mail addresses, etc.

- Useful information could also be gathered from the corporate website: information including e-mail addresses, directory paths (HTML, cgi, ASP, etc). Some corporate websites include user directories listing each users name, responsibility, and e-mail address as well.

3.1.2 Enumeration

In this phase, the information gathered from the reconnaissance phase will be used to gain specific information about each component in order to plan the actual attack - the components include such details as user and naming structure, versions of OS and applications running on servers, types of routers and their OS version.

During the enumeration phase we will try to establishing/mapping out the network using :

-*nslookup/dig/sam spade*: to get as much information about/from the DNS server as possible. We will attempt a zone transfer: this would provide the names and IP address of each host on the GIAC Enterprises network. We will also attempt to identify the specific version of BIND of nameservers, given that BIND has historically been severely affected by bugs. Nslookup and dig are in the BIND distribution, Sam spade is available at http://www.blighty.com/products/spade/.

-*traceroute*: This utility can be used to gather information about the various hops that a packet takes from source to destination. It is useful to identify intervening routers and firewalls.

-*nmap*: This very important/useful tool can be used to scan a network to determine what services are running on each host, in addition, nmap will fingerprint a host to determine the OS version. You can download nmap from http://www.insecure.org/nmap/index.html.

-*Telnet/netcat*: the telnet utility can be used to gather information about various services that may be running on a particular host by "banner grabbing". The netcat utility is a better tool for "banner grabbing", as it can be automated.

-*Hping2:* can be used to forge packets with arbitrary values in almost every field (Es: TTL 1), trying to circumvent the firewall. It is available at http://www.kyuzz.org/antirez/hping/

3.1.3 Vulnerability Mapping

This phase uses the information gathered above along with the publicly-available vulnerabilities list from:

1. http://www.cert.org
2. http://xforce.iss.net
3. http://advice.networkice.com/advice/
4. vendor security alerts
5. Bugtraq at http://www.securityfocus.com/
6. http://cve.mitre.org

With the information gathered here, it is possible to make educated guesses, and to actually use attacks that are likely to succeed.

3.1.4 Condiderations and risks

There are a number of administrative and technical considerations that must be taken cognizance of in executing the Audit:

1. Legal considerations: to prevent a lawsuit, it is advisable to have a written contract stating the risks associated with our activity and providing permission to perform all the attacks we able to launch against the identified targets.
2. Technical considerations: it could happen that servers, hosts, routers and other networking infrastructure can be irreparably compromised by this kind of activity. A disaster recovery plan should be available prior to the implementation of the assessment.
3. False positives: Some tools may result in false positives and therefore the information gathered may not be that useful. The use of multiple tools is highly reccomended.
4. Risk of an actual attack during the assessment.
5. Shifts: Typically the traffic patterns on network operations vary largely between periods of high and moderate or low usage. Therefore it would be necessary to sample both shifts (ES: 9AM-5PM, 6PM-7AM).
6. Level of Effort: Depending on the complexity of the network, the level of effort in performing the audit can vary. Basically, we can produce the following estimates:

    - Recon phase could typically take 1 work day
    - Enumeration phase would typically take 3 days
    - Vulnerability Mapping: typically take 3 days
    - Implementation Stage: the levels of effort here will vary based on what is found out in the planning stage.

3.2 Implementation of Assessment

It is important to point out that the effectiveness of a perimeter protection is only as strong as the weakest point. Every component of an infrastructure like hosts, users, servers, services, etc. could be abused and provide points of entry into a Network.

Our first goal would be to establish that the external router and the firewall actually implement the security policies it is supposed to. To do this, it would be necessary to set up a machine on an external network which would target the internal network with various packets and packet types.

Another machine inside the network would be required to sniff for packets that make it through that should not have. This in conjunction with the firewalls logs will help determine the effectiveness of the firewall.

Next, the roles would be reversed: a host from within the network would be used to target outside the network with legitimate and illegitimate packets and the sniffer would track the results.

The second task includes launching an actual attack over and above the firewall based on the information collected during the planning phase.

3.2.1 Auditing the router

In order to accomplish this task, the following actions must be performed:

1. Port scan the router to see if there are open ports that should not be. Use nmap/strobe/hping2.
2. telnet/ssh the router from various external locations to see if remote administrative access is blocked
3. verify that no snmp server is active
4. verify no finger service is offered
5. verify no tcp and udp small services are offered (ES: telnet router 7, telnet router 9, telnet router 11, telnet router 13, telnet router 19)
6. verify that no http service is offered by the router (ES: *telnet router 80*)
7. verify that the router screened all such accesses (issue the *sho ip access list* command on the router console)
8. verify in the syslog server that all accesses have been logged

3.2.2 Auditing the firewall setup

In accomplishing this goal, we use another paper written by Lance Spitzner as a guide. This paper is available at: http://www.enteract.com/~lspitz/audit.html.

We move along the following lines:

1. define what we expect
2. test the firewall itself
3. test the firewall rulebase

The first statement mandates that we understand what our firewall has to do, and this is described in the security policy. Only after this phase, we can compare results with expectations.

The second item states that the firewall itself should be secure: no one from the outside can access or modify the firewall. Our firewall should be an armored bastion host with as few services as possible started on it. To verify this condition, port scan the firewall from every network interface with icmp, udp and tcp packets looking for open doors. A properly configured firewall should have no open ports. Lastly, we have to verify that the firewall drops all the traffic that is not permitted. This is done scanning every network from every other network to see what packets can and cannot get through the firewall. This scan can take a long time, since our screening rules drop traffic with no response

To determine which TCP ports are open on the firewall we issue the following command from a system located between the router and the firewall:

```
# nmap -vv -O -o fw.out x1.x2.x3.3
```

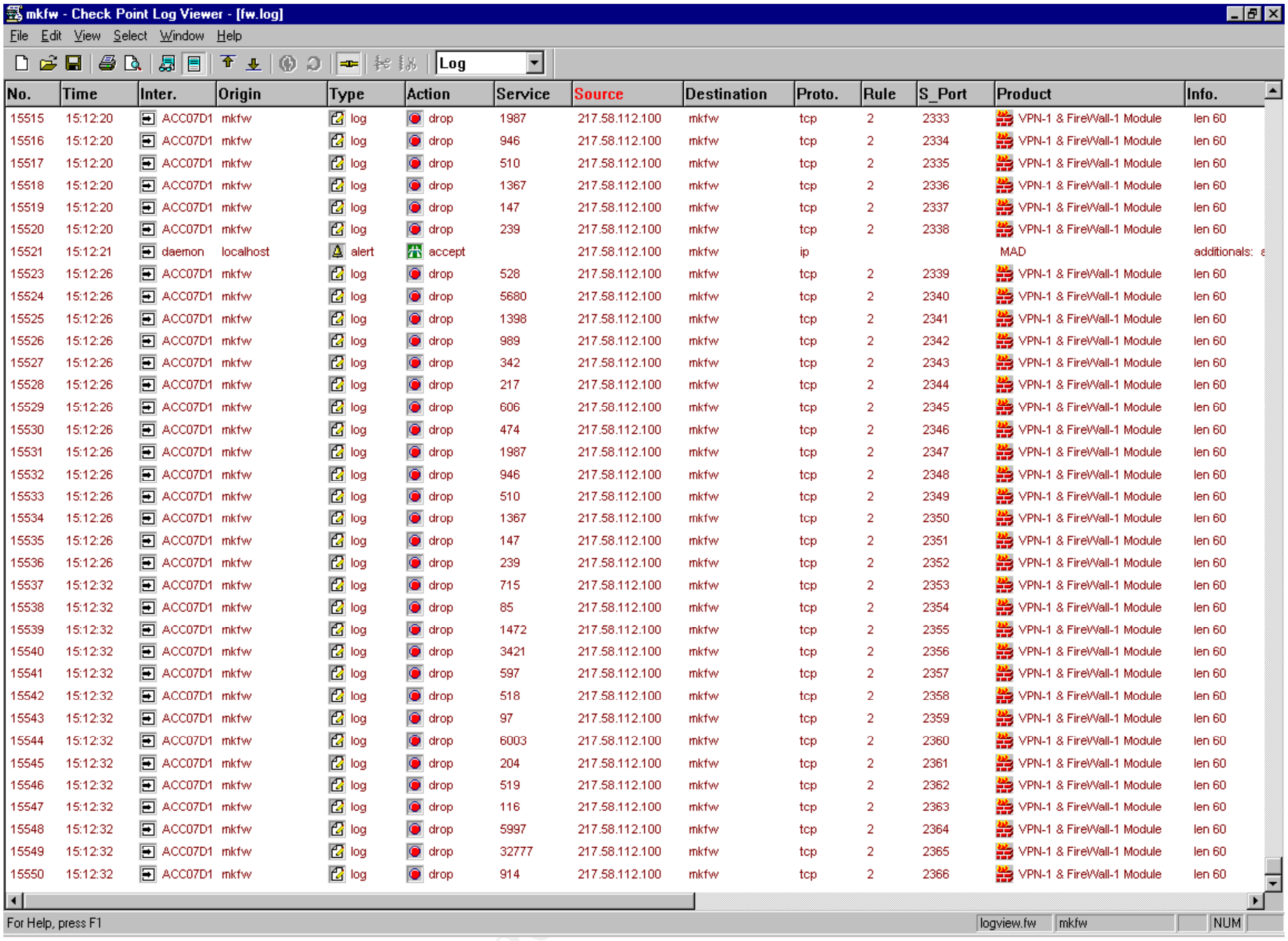
```
Starting nmap V. 2.53 by fyodor@insecure.org ( www.insecure.org/nmap/ )
No tcp,udp, or ICMP scantype specified, assuming vanilla tcp connect() scan. Use -sP if you
really don't want to portscan (and just want to see what hosts are up).
Initiating TCP connect() scan against  (x1.x2.x3.3)
Adding TCP port 265 (state open).
Adding TCP port 264 (state open).
The TCP connect scan took 202 seconds to scan 1523 ports.
For OSScan assuming that port 264 is open and port 40042 is closed and neither are firewalled
Interesting ports on  (217.58.112.245):
(The 1521 ports scanned but not shown below are in state: filtered)
Port       State       Service
264/tcp    open        bgmp
265/tcp    open        unknown

TCP Sequence Prediction: Class=trivial time dependency
                         Difficulty=3 (Trivial joke)

Sequence numbers: 49E9A 49EAC 49EBF 49ECB 49ED9 49EE8
Remote operating system guess: Windows NT4 / Win95 / Win98
OS Fingerprint:
TSeq(Class=TD%gcd=1%SI=3)
T1(Resp=Y%DF=Y%W=2017%ACK=S++%Flags=AS%Ops=M)
T2(Resp=Y%DF=N%W=0%ACK=S%Flags=AR%Ops=)
T3(Resp=Y%DF=Y%W=2017%ACK=S++%Flags=AS%Ops=M)
T4(Resp=Y%DF=N%W=0%ACK=O%Flags=R%Ops=)
T5(Resp=N)
T6(Resp=N)
T7(Resp=N)
PU(Resp=N)
Nmap run completed -- 1 IP address (1 host up) scanned in 235 seconds
```

NMAP found 2 open ports on FW-1: 264 and 265. During this scan we checked the Enable VPN-1 & Firewall-1 Control Connection box in the Policy Editor.

Disabling the Control connections shows a better result:

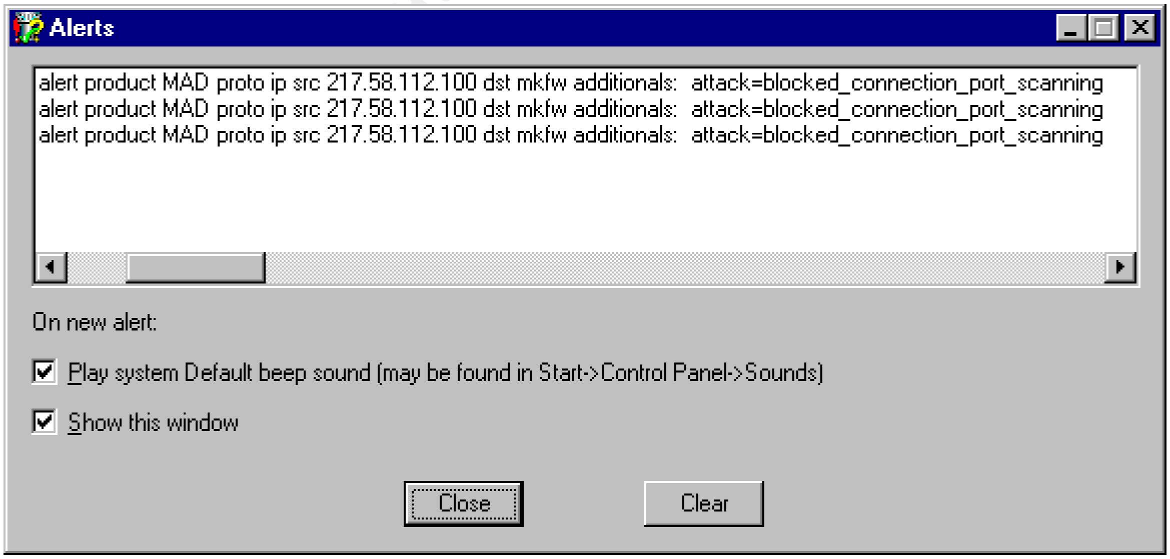
```
# nmap -vv -O -P0 -TAggressive x1.x2.x3.3

Starting nmap V. 2.53 by fyodor@insecure.org ( www.insecure.org/nmap/)
No tcp,udp, or ICMP scantype specified, assuming vanilla tcp connect() scan. Use -sP if you
really don't want to portscan (and just want to see what hosts are up).
Initiating TCP connect() scan against  (x1.x2.x3.3)
Skipping host   (x1.x2.x3.3) due to host timeout
Nmap run completed -- 1 IP address (1 host up) scanned in 300 seconds
```

The scans were detected by FW-1, and raised an alert window, as show in the following figures:


mkfw - Check Point Log Viewer - [fw.log]
File Edit View Select Window Help
Log
For Help, press F1
logview.fw
mkfw
NUM

No.	Time	Inter.	Origin	Type	Action	Service	Source	Destination	Proto.	Rule	S_Port	Product	Info.
15515	15:12:20	ACC07D1	mkfw	log	drop	1987	217.58.112.100	mkfw	tcp	2	2333	VPN-1 & FireWall-1 Module	len 60
15516	15:12:20	ACC07D1	mkfw	log	drop	946	217.58.112.100	mkfw	tcp	2	2334	VPN-1 & FireWall-1 Module	len 60
15517	15:12:20	ACC07D1	mkfw	log	drop	510	217.58.112.100	mkfw	tcp	2	2335	VPN-1 & FireWall-1 Module	len 60
15518	15:12:20	ACC07D1	mkfw	log	drop	1367	217.58.112.100	mkfw	tcp	2	2336	VPN-1 & FireWall-1 Module	len 60
15519	15:12:20	ACC07D1	mkfw	log	drop	147	217.58.112.100	mkfw	tcp	2	2337	VPN-1 & FireWall-1 Module	len 60
15520	15:12:20	ACC07D1	mkfw	log	drop	239	217.58.112.100	mkfw	tcp	2	2338	VPN-1 & FireWall-1 Module	len 60
15521	15:12:21	daemon	localhost	alert	accept		217.58.112.100	mkfw	ip			MAD	additionals: a
15523	15:12:26	ACC07D1	mkfw	log	drop	528	217.58.112.100	mkfw	tcp	2	2339	VPN-1 & FireWall-1 Module	len 60
15524	15:12:26	ACC07D1	mkfw	log	drop	5680	217.58.112.100	mkfw	tcp	2	2340	VPN-1 & FireWall-1 Module	len 60
15525	15:12:26	ACC07D1	mkfw	log	drop	1398	217.58.112.100	mkfw	tcp	2	2341	VPN-1 & FireWall-1 Module	len 60
15526	15:12:26	ACC07D1	mkfw	log	drop	989	217.58.112.100	mkfw	tcp	2	2342	VPN-1 & FireWall-1 Module	len 60
15527	15:12:26	ACC07D1	mkfw	log	drop	342	217.58.112.100	mkfw	tcp	2	2343	VPN-1 & FireWall-1 Module	len 60
15528	15:12:26	ACC07D1	mkfw	log	drop	217	217.58.112.100	mkfw	tcp	2	2344	VPN-1 & FireWall-1 Module	len 60
15529	15:12:26	ACC07D1	mkfw	log	drop	606	217.58.112.100	mkfw	tcp	2	2345	VPN-1 & FireWall-1 Module	len 60
15530	15:12:26	ACC07D1	mkfw	log	drop	474	217.58.112.100	mkfw	tcp	2	2346	VPN-1 & FireWall-1 Module	len 60
15531	15:12:26	ACC07D1	mkfw	log	drop	1987	217.58.112.100	mkfw	tcp	2	2347	VPN-1 & FireWall-1 Module	len 60
15532	15:12:26	ACC07D1	mkfw	log	drop	946	217.58.112.100	mkfw	tcp	2	2348	VPN-1 & FireWall-1 Module	len 60
15533	15:12:26	ACC07D1	mkfw	log	drop	510	217.58.112.100	mkfw	tcp	2	2349	VPN-1 & FireWall-1 Module	len 60
15534	15:12:26	ACC07D1	mkfw	log	drop	1367	217.58.112.100	mkfw	tcp	2	2350	VPN-1 & FireWall-1 Module	len 60
15535	15:12:26	ACC07D1	mkfw	log	drop	147	217.58.112.100	mkfw	tcp	2	2351	VPN-1 & FireWall-1 Module	len 60
15536	15:12:26	ACC07D1	mkfw	log	drop	239	217.58.112.100	mkfw	tcp	2	2352	VPN-1 & FireWall-1 Module	len 60
15537	15:12:32	ACC07D1	mkfw	log	drop	715	217.58.112.100	mkfw	tcp	2	2353	VPN-1 & FireWall-1 Module	len 60
15538	15:12:32	ACC07D1	mkfw	log	drop	85	217.58.112.100	mkfw	tcp	2	2354	VPN-1 & FireWall-1 Module	len 60
15539	15:12:32	ACC07D1	mkfw	log	drop	1472	217.58.112.100	mkfw	tcp	2	2355	VPN-1 & FireWall-1 Module	len 60
15540	15:12:32	ACC07D1	mkfw	log	drop	3421	217.58.112.100	mkfw	tcp	2	2356	VPN-1 & FireWall-1 Module	len 60
15541	15:12:32	ACC07D1	mkfw	log	drop	597	217.58.112.100	mkfw	tcp	2	2357	VPN-1 & FireWall-1 Module	len 60
15542	15:12:32	ACC07D1	mkfw	log	drop	518	217.58.112.100	mkfw	tcp	2	2358	VPN-1 & FireWall-1 Module	len 60
15543	15:12:32	ACC07D1	mkfw	log	drop	97	217.58.112.100	mkfw	tcp	2	2359	VPN-1 & FireWall-1 Module	len 60
15544	15:12:32	ACC07D1	mkfw	log	drop	6003	217.58.112.100	mkfw	tcp	2	2360	VPN-1 & FireWall-1 Module	len 60
15545	15:12:32	ACC07D1	mkfw	log	drop	204	217.58.112.100	mkfw	tcp	2	2361	VPN-1 & FireWall-1 Module	len 60
15546	15:12:32	ACC07D1	mkfw	log	drop	519	217.58.112.100	mkfw	tcp	2	2362	VPN-1 & FireWall-1 Module	len 60
15547	15:12:32	ACC07D1	mkfw	log	drop	116	217.58.112.100	mkfw	tcp	2	2363	VPN-1 & FireWall-1 Module	len 60
15548	15:12:32	ACC07D1	mkfw	log	drop	5997	217.58.112.100	mkfw	tcp	2	2364	VPN-1 & FireWall-1 Module	len 60
15549	15:12:32	ACC07D1	mkfw	log	drop	32777	217.58.112.100	mkfw	tcp	2	2365	VPN-1 & FireWall-1 Module	len 60
15550	15:12:32	ACC07D1	mkfw	log	drop	914	217.58.112.100	mkfw	tcp	2	2366	VPN-1 & FireWall-1 Module	len 60

Fig. 6 – Log output



```
alert product MAD proto ip src 217.58.112.100 dst mkfw additionals:  attack=blocked_connection_port_scanning
alert product MAD proto ip src 217.58.112.100 dst mkfw additionals:  attack=blocked_connection_port_scanning
alert product MAD proto ip src 217.58.112.100 dst mkfw additionals:  attack=blocked_connection_port_scanning
```

On new alert:

☑ Play system Default beep sound (may be found in Start->Control Panel->Sounds)

☑ Show this window

Fig. 7 – Alert window

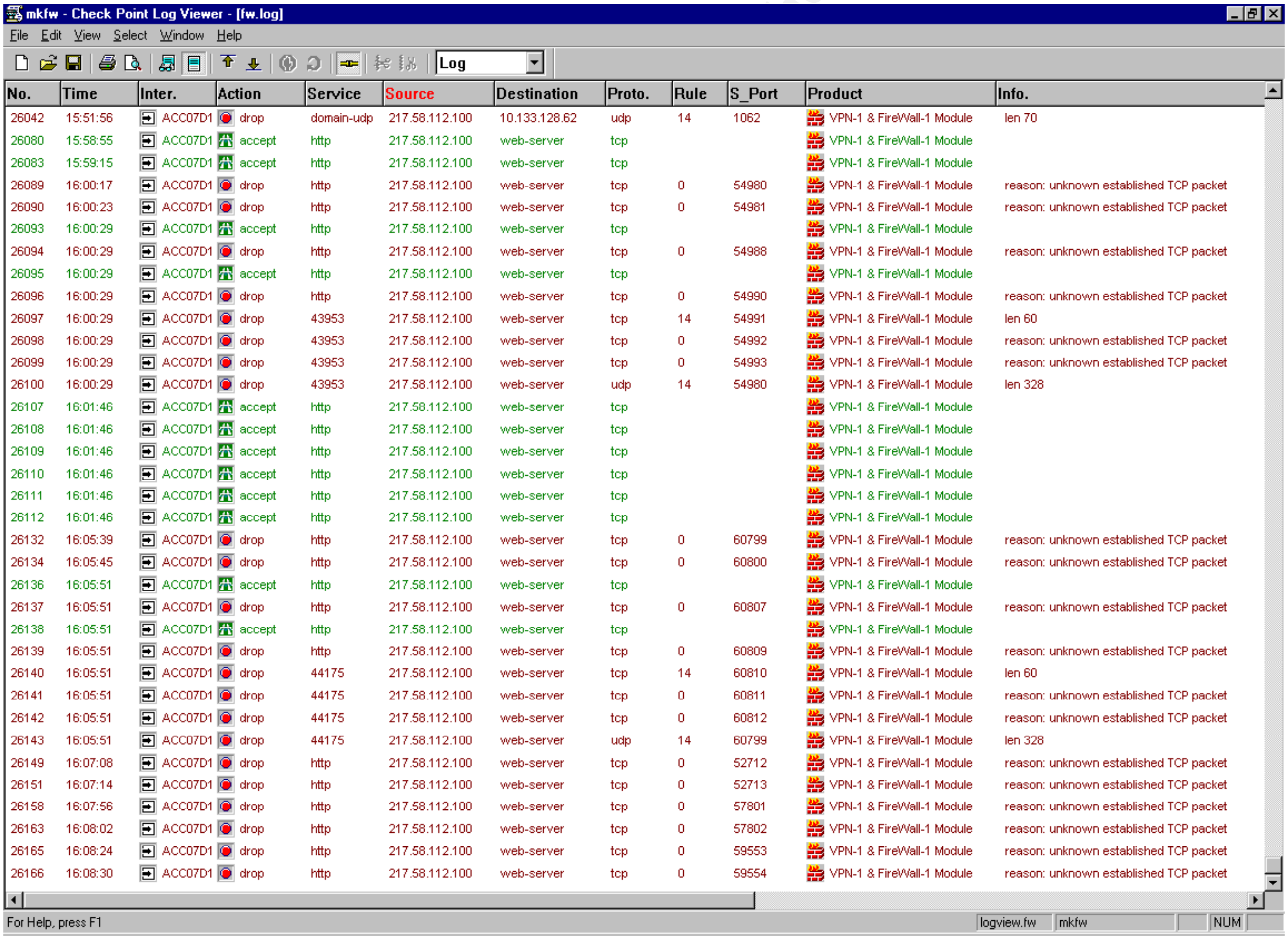
To determine which TCP ports are filtered by the firewall, we stealth scan a protected system. We use NULL, FIN and XMAS stealth scans, in order to verify if the firewall can detect and block such activities. We set the source port to 80, so our packets look like replies from a remote www server (port 53 could be used as well, to masquerade this activity as dns lookup/reply traffic) .We issue these commands from the same remote system:

```
[root@salvo ~]# nmap -vv -sN -O -P0 x1.x2.x3.66 -p 80

Starting nmap V. 2.53 by fyodor@insecure.org ( www.insecure.org/nmap/ )
Initiating FIN,NULL, UDP, or Xmas stealth scan against  (x1.x2.x3.66)
The UDP or stealth FIN/NULL/XMAS scan took 12 seconds to scan 1 ports.
For OSScan assuming that port 80 is open and port 43953 is closed and neither are firewalled
Interesting ports on  (x1.x2.x3.66):
Port       State       Service
80/tcp     open        http

TCP Sequence Prediction: Class=trivial time dependency
                         Difficulty=2 (Trivial joke)

Sequence numbers: DABE DAD0 DAE0 DAE6 DAF2 DB04
Remote operating system guess: NT Server 4.0 SP5 running Checkpoint Firewall-1
OS Fingerprint:
TSeq(Class=TD%gcd=2%SI=2)
T1(Resp=Y%DF=Y%W=2017%ACK=S++%Flags=AS%Ops=M)
T2(Resp=N)
T3(Resp=Y%DF=Y%W=2017%ACK=S++%Flags=AS%Ops=M)
T4(Resp=N)
T5(Resp=N)
T6(Resp=N)
T7(Resp=N)
PU(Resp=N)
Nmap run completed -- 1 IP address (1 host up) scanned in 90 seconds


[root@salvo ~]# nmap -vv -sN  -P0 x1.x2.x3.66 -p 80
Starting nmap V. 2.53 by fyodor@insecure.org ( www.insecure.org/nmap/ )
Initiating FIN,NULL, UDP, or Xmas stealth scan against  (x1.x2.x3.66)
The UDP or stealth FIN/NULL/XMAS scan took 12 seconds to scan 1 ports.
Interesting ports on  (x1.x2.x3.66):
Port       State       Service
80/tcp     open        http

Nmap run completed -- 1 IP address (1 host up) scanned in 12 seconds

[root@salvo ~]# nmap -vv -sX -P0 x1.x2.x3.66 -p 80
Starting nmap V. 2.53 by fyodor@insecure.org ( www.insecure.org/nmap/ )
Initiating FIN,NULL, UDP, or Xmas stealth scan against  (x1.x2.x3.66)
The UDP or stealth FIN/NULL/XMAS scan took 12 seconds to scan 1 ports.
Interesting ports on  (x1.x2.x3.66):
Port       State       Service
80/tcp     open        http

Nmap run completed -- 1 IP address (1 host up) scanned in 12 seconds

[root@salvo ~]# nmap -vv -sF -P0 x1.x2.x3.66 -p 80
Starting nmap V. 2.53 by fyodor@insecure.org ( www.insecure.org/nmap/ )
Initiating FIN,NULL, UDP, or Xmas stealth scan against  (x1.x2.x3.66)
```

```
The UDP or stealth FIN/NULL/XMAS scan took 12 seconds to scan 1 ports.
Interesting ports on  (x1.x2.x3.66):
Port       State       Service
80/tcp     open        http

Nmap run completed -- 1 IP address (1 host up) scanned in 12 seconds
```

All these scans are blocked by FW-1, as the following figure shows, however it should be noted that the FW-1 implementation on WNT has a serious problem with the sequence number generation that is quite trivial to hijack. During the first Null scan, while trying to determine the OS of the protected Web server, we could determine the firewall configuration at GIAC.

We therefore recommend that in the near future, this firewall would be substituted by a Nokia machine with the IPSO version of FW-1.



No.	Time	Inter.	Action	Service	Source	Destination	Proto.	Rule	S_Port	Product	Info.
26042	15:51:56	ACC07D1	drop	domain-udp	217.58.112.100	10.133.128.62	udp	14	1062	VPN-1 & FireWall-1 Module	len 70
26080	15:58:55	ACC07D1	accept	http	217.58.112.100	web-server	tcp			VPN-1 & FireWall-1 Module	
26083	15:59:15	ACC07D1	accept	http	217.58.112.100	web-server	tcp			VPN-1 & FireWall-1 Module	
26089	16:00:17	ACC07D1	drop	http	217.58.112.100	web-server	tcp	0	54980	VPN-1 & FireWall-1 Module	reason: unknown established TCP packet
26090	16:00:23	ACC07D1	drop	http	217.58.112.100	web-server	tcp	0	54981	VPN-1 & FireWall-1 Module	reason: unknown established TCP packet
26093	16:00:29	ACC07D1	accept	http	217.58.112.100	web-server	tcp			VPN-1 & FireWall-1 Module	
26094	16:00:29	ACC07D1	drop	http	217.58.112.100	web-server	tcp	0	54988	VPN-1 & FireWall-1 Module	reason: unknown established TCP packet
26095	16:00:29	ACC07D1	accept	http	217.58.112.100	web-server	tcp			VPN-1 & FireWall-1 Module	
26096	16:00:29	ACC07D1	drop	http	217.58.112.100	web-server	tcp	0	54990	VPN-1 & FireWall-1 Module	reason: unknown established TCP packet
26097	16:00:29	ACC07D1	drop	43953	217.58.112.100	web-server	tcp	14	54991	VPN-1 & FireWall-1 Module	len 60
26098	16:00:29	ACC07D1	drop	43953	217.58.112.100	web-server	tcp	0	54992	VPN-1 & FireWall-1 Module	reason: unknown established TCP packet
26099	16:00:29	ACC07D1	drop	43953	217.58.112.100	web-server	tcp	0	54993	VPN-1 & FireWall-1 Module	reason: unknown established TCP packet
26100	16:00:29	ACC07D1	drop	43953	217.58.112.100	web-server	udp	14	54980	VPN-1 & FireWall-1 Module	len 328
26107	16:01:46	ACC07D1	accept	http	217.58.112.100	web-server	tcp			VPN-1 & FireWall-1 Module	
26108	16:01:46	ACC07D1	accept	http	217.58.112.100	web-server	tcp			VPN-1 & FireWall-1 Module	
26109	16:01:46	ACC07D1	accept	http	217.58.112.100	web-server	tcp			VPN-1 & FireWall-1 Module	
26110	16:01:46	ACC07D1	accept	http	217.58.112.100	web-server	tcp			VPN-1 & FireWall-1 Module	
26111	16:01:46	ACC07D1	accept	http	217.58.112.100	web-server	tcp			VPN-1 & FireWall-1 Module	
26112	16:01:46	ACC07D1	accept	http	217.58.112.100	web-server	tcp			VPN-1 & FireWall-1 Module	
26132	16:05:39	ACC07D1	drop	http	217.58.112.100	web-server	tcp	0	60799	VPN-1 & FireWall-1 Module	reason: unknown established TCP packet
26134	16:05:45	ACC07D1	drop	http	217.58.112.100	web-server	tcp	0	60800	VPN-1 & FireWall-1 Module	reason: unknown established TCP packet
26136	16:05:51	ACC07D1	accept	http	217.58.112.100	web-server	tcp			VPN-1 & FireWall-1 Module	
26137	16:05:51	ACC07D1	drop	http	217.58.112.100	web-server	tcp	0	60807	VPN-1 & FireWall-1 Module	reason: unknown established TCP packet
26138	16:05:51	ACC07D1	accept	http	217.58.112.100	web-server	tcp			VPN-1 & FireWall-1 Module	
26139	16:05:51	ACC07D1	drop	http	217.58.112.100	web-server	tcp	0	60809	VPN-1 & FireWall-1 Module	reason: unknown established TCP packet
26140	16:05:51	ACC07D1	drop	44175	217.58.112.100	web-server	tcp	14	60810	VPN-1 & FireWall-1 Module	len 60
26141	16:05:51	ACC07D1	drop	44175	217.58.112.100	web-server	tcp	0	60811	VPN-1 & FireWall-1 Module	reason: unknown established TCP packet
26142	16:05:51	ACC07D1	drop	44175	217.58.112.100	web-server	tcp	0	60812	VPN-1 & FireWall-1 Module	reason: unknown established TCP packet
26143	16:05:51	ACC07D1	drop	44175	217.58.112.100	web-server	udp	14	60799	VPN-1 & FireWall-1 Module	len 328
26149	16:07:08	ACC07D1	drop	http	217.58.112.100	web-server	tcp	0	52712	VPN-1 & FireWall-1 Module	reason: unknown established TCP packet
26151	16:07:14	ACC07D1	drop	http	217.58.112.100	web-server	tcp	0	52713	VPN-1 & FireWall-1 Module	reason: unknown established TCP packet
26158	16:07:56	ACC07D1	drop	http	217.58.112.100	web-server	tcp	0	57801	VPN-1 & FireWall-1 Module	reason: unknown established TCP packet
26163	16:08:02	ACC07D1	drop	http	217.58.112.100	web-server	tcp	0	57802	VPN-1 & FireWall-1 Module	reason: unknown established TCP packet
26165	16:08:24	ACC07D1	drop	http	217.58.112.100	web-server	tcp	0	59553	VPN-1 & FireWall-1 Module	reason: unknown established TCP packet
26166	16:08:30	ACC07D1	drop	http	217.58.112.100	web-server	tcp	0	59554	VPN-1 & FireWall-1 Module	reason: unknown established TCP packet

Fig.8 – Stealth scans on the Web server

For more information on nmap options and usage, read Fyodor's docs at http://www.insecure.org/nmap/nmap_documentation.html.

Using hping2, a tool developped by Salvatore Sanfilippo, we perform two additional scans: a SYN+FIN scan and a reserved bit scan. We issue the following commands:

```
# hping2 -S -F x1.x2.x3.66 (here we set the SYN and the Fin bits, an invalid combination)
# hping2 -X -Y x1.x2.x3.66 (here we set the two reserved flag bits)
```

Another method can be used to determine the access list on the firewall, and is based on a technology known as firewalking. An excellent paper on firewalking can be found at http://www.es2.net/research/firewalk/.

This method depends onthe fact that the firewall generates a ICMP TTL expired error message when it receives a packet with a TTL of 1, and has to be forwarded. So we set packets with TTL 1 and try to send them to a protected system. When such packets reach the firewall two events can occur:

1. If the packet can be forwarded, its TTL is decremented, so TTL goes to 0 and the packet is dropped with an error message.
2. If the packet is dropped, no response will be generated.

To firewalk our network, we cannot use nmap because it doesn't allow to set the TTL value, so we have to use hping2, with the drawback that only one port at a time can be scanned.

3.2.3 Audit the protected networks

Once we have determined what the firewall allows through, we have to define what threat that poses. We have to audit resources behind the firewall. Our goal is to determine what potential vulnerabilities exist for the accessible resources.

Our best friends in accomplishing this duty are:

1. Nessus,available at http://www.nessus.org/, one of the best free vulnerability scanners
2. Sara
3. Saint, available at http://www.wwdsi.com/, an excellent tool for the auditing of networks and hosts
4. ISS,
5. Winfingerprint, available at http://www.technotronic.com/winfingerprint, Enumerates NetBIOS Shares, Users, Groups, and Services
6. CISCO Secure Scanner, aka Netsonar
7. LANguard Network Scanner, a freeware security scanner available at http://www.languard.com/languard/lantools.htm It scans entire networks and provides NETBIOS information for each computer such as hostname, shares, logged on user name. It does OS detection, tests password strength, detects registry issues and much more. Reports are outputted in HTML.

The last activity we may want to consider is running a DDOS attack against GIAC firewall. Such an activity, however, must be coordinated well in advance. Nevertheless, we can can attack GIAC firewall with Nessus that has a large database of Denial of Service attacks in order to test that the protected networks are not vulnerable to such activities.

3.2.4 Review logs

After the completion of all these activities, we have to review the firewall logs. We must be sure that the firewall detected all our scans and raised all the expected alarms.

Here we analyze two such log alarms.

```
From root@gwpa.giac.it Thu Aug 27 17:00:19 2001
Date: Mon, 27 Aug 2001 17:00:10 +0200 (CEST)
From: system PRIVILEGED account <root>
To: root
Subject: gwpa.giac.it 08/27/01:17.00 ACTIVE SYSTEM ATTACK!


Active System Attack Alerts
=-=-=-=-=-=-=-=-=-=-=-=-=-=
Aug 27 16:00:17 router 8928: *Mar  6 06:35:29.136: %SEC-6-IPACCESSLOGP: list 111 denied tcp
217.58.118.130(3174) -> x1.x2.x3.0(80), 1 packet
Aug 27 16:00:57 router 8929: *Mar  6 06:36:09.032: %SEC-6-IPACCESSLOGP: list 111 denied tcp
217.58.207.195(4054) -> x1.x2.x3.65(80), 1 packet
Aug 27 16:01:42 router 8930: *Mar  6 06:36:53.216: %SEC-6-IPACCESSLOGP: list 111 denied tcp
217.58.1.150(3578) -> x1.x2.x3.63(80), 1 packet
Aug 27 16:01:47 router 8931: *Mar  6 06:36:58.652: %SEC-6-IPACCESSLOGP: list 111 denied tcp
217.58.118.130(3605) -> x1.x2.x3.246(80), 1 packet
Aug 27 16:02:01 router 8932: *Mar  6 06:37:12.272: %SEC-6-IPACCESSLOGP: list 111 denied tcp
217.58.85.98(3880) -> x1.x2.x3.246(80), 1 packet
Aug 27 16:02:13 router 8933: *Mar  6 06:37:24.184: %SEC-6-IPACCESSLOGP: list 111 denied tcp
217.58.207.195(4600) -> x1.x2.x3.66(80), 2 packets
Aug 27 17:00:00 router 9061: *Mar  6 07:35:12.208: %SEC-6-IPACCESSLOGP: list 111 denied tcp
217.58.85.98(3818) -> x1.x2.x3.63(80), 1 packet
```

These are attempts to access non existant web servers on the service net. They are blocked by the router.

```
08/27-16:14:04.511142  [**] [1:1243:1] WEB-IIS ISAPI .ida attempt [**] [Classification:
Attempted Administrator Privilege Gain] [Priority: 10] {TCP} 217.58.114.250:3868 ->
x1.x2.x3.66:80
08/27-16:14:04.533603  [**] [1:1002:1] WEB-IIS cmd.exe access [**] [Classification: Attempted
User Privilege Gain] [Priority: 8] {TCP} 217.58.114.250:3868 -> x1.x2.x3.66:80
08/27-16:21:06.360751  [**] [1:1243:1] WEB-IIS ISAPI .ida attempt [**] [Classification:
Attempted Administrator Privilege Gain] [Priority: 10] {TCP} 217.58.211.106:4802 ->
x1.x2.x3.66:80
08/27-16:21:06.384189  [**] [1:1002:1] WEB-IIS cmd.exe access [**] [Classification: Attempted
User Privilege Gain] [Priority: 8] {TCP} 217.58.211.106:4802 -> x1.x2.x3.66:80
```

These are attempts to subvert the GIAC web server using attack patterns like those generated by Red CodeII compromised systems. Really our net was flooded by such attacks during our assessment. Such log was generated by snort.

```
From root@gwpa.giac.it Thu Aug 30 12:00:38 2001
Date: Thu, 30 Aug 2001 12:00:09 +0200 (CEST)
From: system PRIVILEGED account <root>
To: root
Subject: gwpa.giac.it 08/30/01:12.00 ACTIVE SYSTEM ATTACK!


Active System Attack Alerts
```

```
=-=-=-=-=-=-=-=-=-=-=-=

Aug 30 11:00:43 router 19153: *Mar  9 01:36:00.763: %SEC-6-IPACCESSLOGP: list 111 denied tcp
217.58.108.98(1227) -> x1.x2.x3.0(80), 1 packet
Aug 30 11:00:55 router 19154: *Mar  9 01:36:12.823: %SEC-6-IPACCESSLOGP: list 111 denied tcp
217.58.114.251(3572) -> x1.x2.x3.244(80), 1 packet
Aug 30 11:00:58 router 19155: *Mar  9 01:36:16.275: %SEC-6-IPACCESSLOGP: list 111 denied tcp
217.58.114.251(1417) -> x1.x2.x3.244(80), 2 packets
Aug 30 11:01:37 router 19156: *Mar  9 01:36:54.859: %SEC-6-IPACCESSLOGP: list 111 denied tcp
217.58.108.98(3767) -> x1.x2.x3.3(80), 1 packet
Aug 30 11:01:54 router 19157: *Mar  9 01:37:12.283: %SEC-6-IPACCESSLOGP: list 111 denied tcp
217.58.85.58(4893) -> x1.x2.x3.63(80), 1 packet
Aug 30 11:02:28 router 19158: *Mar  9 01:37:45.435: %SEC-6-IPACCESSLOGP: list 111 denied tcp
217.58.114.250(3639) -> x1.x2.x3.67(80), 1 packet
Aug 30 11:02:59 router 19159: *Mar  9 01:38:16.359: %SEC-6-IPACCESSLOGP: list 111 denied tcp
217.58.207.195(3263) -> x1.x2.x3.250(80), 2 packets
Aug 30 11:07:59 router 19173: *Mar  9 01:43:16.583: %SEC-6-IPACCESSLOGP: list 111 denied tcp
217.58.114.250(3639) -> x1.x2.x3.67(80), 2 packets
```

Other Red CodeII scans from compromised machines

```
Aug 30 11:08:11 router 19174: *Mar  9 01:43:28.835: %SEC-6-IPACCESSLOGP: list 111 denied tcp
192.168.1.3(1234) -> x1.x2.x3.3(2), 1 packet
Aug 30 11:09:11 router 19182: *Mar  9 01:44:28.755: %SEC-6-IPACCESSLOGP: list 111 denied tcp
192.168.1.3(1234) -> x1.x2.x3.3(6), 1 packet
Aug 30 11:09:26 router 19183: *Mar  9 01:44:43.747: %SEC-6-IPACCESSLOGP: list 111 denied tcp
192.168.1.3(1234) -> x1.x2.x3.3(7), 1 packet
Aug 30 11:09:41 router 19184: *Mar  9 01:44:58.915: %SEC-6-IPACCESSLOGP: list 111 denied tcp
192.168.1.3(1234) -> x1.x2.x3.3(8), 1 packet
Aug 30 11:10:26 router 19190: *Mar  9 01:45:43.843: %SEC-6-IPACCESSLOGP: list 111 denied tcp
192.168.1.3(1234) -> x1.x2.x3.3(11), 1 packet
Aug 30 11:10:41 router 19191: *Mar  9 01:45:58.871: %SEC-6-IPACCESSLOGP: list 111 denied tcp
192.168.1.3(1234) -> x1.x2.x3.3(12), 1 packet
Aug 30 11:10:56 router 19192: *Mar  9 01:46:13.907: %SEC-6-IPACCESSLOGP: list 111 denied tcp
192.168.1.3(1234) -> x1.x2.x3.3(13), 1 packet
Aug 30 11:11:11 router 19194: *Mar  9 01:46:28.951: %SEC-6-IPACCESSLOGP: list 111 denied tcp
192.168.1.3(1234) -> x1.x2.x3.3(14), 1 packet
```

This is a scan of the firewall from the Internet with a forged source address and has been blocked by our router access list

```
08/30-11:00:19.395907  [**] [1:1243:1] WEB-IIS ISAPI .ida attempt [**] [Classification:
Attempted Administrator Privilege Gain] [Priority: 10] {TCP} 217.58.207.195:3790 ->
x1.x2.x3.66:80
08/30-11:00:19.414462  [**] [1:1002:1] WEB-IIS cmd.exe access [**] [Classification: Attempted
User Privilege Gain] [Priority: 8] {TCP} 217.58.207.195:3790 -> x1.x2.x3.66:80
08/30-11:01:42.970126  [**] [1:1243:1] WEB-IIS ISAPI .ida attempt [**] [Classification:
Attempted Administrator Privilege Gain] [Priority: 10] {TCP} 217.58.22.122:3927 ->
x1.x2.x3.66:80
08/30-11:01:42.987704  [**] [1:1002:1] WEB-IIS cmd.exe access [**] [Classification: Attempted
User Privilege Gain] [Priority: 8] {TCP} 217.58.22.122:3927 -> x1.x2.x3.66:80
08/30-11:03:24.498446  [**] [1:1243:1] WEB-IIS ISAPI .ida attempt [**] [Classification:
Attempted Administrator Privilege Gain] [Priority: 10] {TCP} 217.58.146.196:54286 ->
x1.x2.x3.66:80
08/30-11:03:24.516025  [**] [1:1002:1] WEB-IIS cmd.exe access [**] [Classification: Attempted
User Privilege Gain] [Priority: 8] {TCP} 217.58.146.196:54286 -> x1.x2.x3.66:80
```

These are attempts to subvert the GIAC web server using attack patterns like those generated by Red CodeII compromised systems. Really our net was flooded by such attacks during our assessment. Such log was generated by snort.

```
Aug 30 13:14:04 router 19923: *Mar  9 03:49:21.675: %SEC-6-IPACCESSLOGP: list 111 denied udp
147.163.1.78(58798) -> x1.x2.x3.66(40564), 1 packet
Aug 30 13:14:16 router 19926: *Mar  9 03:49:33.923: %SEC-6-IPACCESSLOGP: list 111 denied udp
147.163.1.78(58798) -> x1.x2.x3.66(38868), 1 packet
```

```
Aug 30 13:14:56 router 19931: *Mar  9 03:50:14.443: %SEC-6-IPACCESSLOGP: list 111 denied udp
147.163.1.78(61245) -> x1.x2.x3.66(43475), 1 packet
Aug 30 13:15:02 router 19933: *Mar  9 03:50:20.023: %SEC-6-IPACCESSLOGP: list 111 denied udp
147.163.1.78(61245) -> x1.x2.x3.66(31647), 1 packet
```

These are packets correctly blocked by our router during our nmap scan

```
08/30-13:16:45.177157  [**] [111:10:1] spp_stream4: STEALTH ACTIVITY (nmap XMAS scan)
detection [**] {TCP} 147.163.1.78:54154 -> x1.x2.x3.66:36669
08/30-13:16:47.197665  [**] [111:9:1] spp_stream4: STEALTH ACTIVITY (NULL scan) detection [**]
{TCP} 147.163.1.78:54149 -> x1.x2.x3.66:80
08/30-13:16:47.201571  [**] [111:10:1] spp_stream4: STEALTH ACTIVITY (nmap XMAS scan)
detection [**] {TCP} 147.163.1.78:54154 -> x1.x2.x3.66:36669
08/30-13:16:51.427157  [**] [111:12:1] spp_stream4: NMAP FINGERPRINT (stateful) detection [**]
{TCP} 147.163.1.78:54151 -> x1.x2.x3.66:80
08/30-13:16:51.430087  [**] [111:10:1] spp_stream4: STEALTH ACTIVITY (nmap XMAS scan)
detection [**] {TCP} 147.163.1.78:54154 -> x1.x2.x3.66:43798
08/30-13:16:53.303134  [**] [111:9:1] spp_stream4: STEALTH ACTIVITY (NULL scan) detection [**]
{TCP} 147.163.1.78:54149 -> x1.x2.x3.66:80
08/30-13:16:53.308993  [**] [111:10:1] spp_stream4: STEALTH ACTIVITY (nmap XMAS scan)
detection [**] {TCP} 147.163.1.78:54154 -> x1.x2.x3.66:43798
08/30-13:16:57.251376  [**] [111:12:1] spp_stream4: NMAP FINGERPRINT (stateful) detection [**]
{TCP} 147.163.1.78:54151 -> x1.x2.x3.66:80
```

These are nmap fingerprint and stealth scans against our www server blocked by the firewall and intercepted by snort sniffing on the external interface

```
09/03-12:58:40.978915  [**] [111:13:1] spp_stream4: STEALTH ACTIVITY (SYN FIN scan) detection
[**] {TCP} 151.39.63.67:2603 -> x1.x2.x3.66:80
09/03-12:58:41.980868  [**] [111:13:1] spp_stream4: STEALTH ACTIVITY (SYN FIN scan) detection
[**] {TCP} 151.39.63.67:2604 -> x1.x2.x3.66:80
09/03-12:58:42.986728  [**] [111:13:1] spp_stream4: STEALTH ACTIVITY (SYN FIN scan) detection
[**] {TCP} 151.39.63.67:2605 -> x1.x2.x3.66:80
09/03-12:58:43.979892  [**] [111:13:1] spp_stream4: STEALTH ACTIVITY (SYN FIN scan) detection
[**] {TCP} 151.39.63.67:2606 -> x1.x2.x3.66:80
```

These are hping2 scans with the SYN and FIN bits both set (hping2 –S –F). They are logged by snort and blocked by FW-1.

3.3 Recommendations and alternatives

The audit performed on the GIAC Enterprises network will provide the Security administrator at GIAC with a report containing some recommendations to improve the security of the site.

In particular, it would contain two suggestions:

1. Use a more robust platform to host the external firewall. WNT has a too weak IP implementation: nmap reported the sequence numbers could be too easily guessed. A better alternative could be to use a Nokia (ES: Nokia IP530) appliance with IPSO (a BSD derivative) and Checkpoint FW-1 on it.
2. Use the IOS firewall feature set on the external router to provide for an additional layer of protection.

IOS Firewall uses a technology that Cisco named CBAC or Context Based Access Control .

CBAC intelligently filters TCP and UDP packets based on application-layer protocol session information. To learn more on CBAC, see http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/120newft/120t/120t5/iosfw2/index.htm

Without CBAC, traffic filtering is limited to access list implementations that examine packets at the network layer, or at most, the transport layer. CBAC examines not only network layer and transport layer information but also examines the application-layer protocol information (such as FTP connection information) to learn about the state of the TCP or UDP session. This allows support of protocols that involve multiple channels created as a result of negotiations in the control channel.

Using CBAC, Java blocking can be configured to filter traffic based on the server address or to completely deny access to Java applets that are not embedded in an archived or compressed file. With Java, you must protect against the risk of users inadvertently downloading destructive applets into your network.

CBAC inspects traffic that travels through the firewall to discover and manage state information for TCP and UDP sessions. This state information is used to create temporary openings in the firewall's access lists to allow return traffic and additional data connections for permissible sessions (sessions that originated from within the protected internal network).

Inspecting packets at the application layer, and maintaining TCP and UDP session information, provides CBAC with the ability to detect and prevent certain types of network attacks such as SYN-flooding.

CBAC inspection helps to protect against DoS attacks in other ways. CBAC inspects packet sequence numbers in TCP connections to see if they are within expected ranges---CBAC drops any suspicious packets. You can also configure CBAC to drop half-open connections, which require firewall processing and memory resources to maintain. Additionally, CBAC can detect unusually high rates of new connections and issue alert messages.

CBAC inspection can help protect against certain DoS attacks involving fragmented IP packets.

CBAC also generates real-time alerts and audit trails based on events tracked by the firewall. Enhanced audit trail features use SYSLOG to track all network transactions; recording time stamps, source host, destination host, ports used, and the total number of transmitted bytes, for advanced, session-based reporting.

The Cisco IOS Firewall offers also intrusion detection technology. The Cisco IOS Firewall's Intrusion Detection System (Cisco IOS IDS) identifies 59 of the most common attacks using signatures to detect patterns of misuse in network traffic.

As an example, the following configuration commands could be used to enable a stateful capability on the router:

```
! enable tcp intercept for public tcp services to reduce SYN flooding
access-list 101 permit tcp any host x1.x2.x3.67 eq smtp
```

```
access-list 101 permit tcp any host x1.x2.x3.66 eq 80
access-list 101 permit tcp any host x1.x2.x3.66 eq 443
!
ip tcp intercept list 101
ip tcp intercept mode intercept
```

CBAC will provide some protection against malicious Java (that is not encapsulated), TCP, UDP , IP fragments, and improper SMTP commands.

```
ip access-list standard javatype
 permit <ip address of known legitimate site>
 deny any
!
ip inspect name cbac http javatype
ip inspect name cbac fragment max 100 timeout 4
ip inspect name cbac mail
ip inspect name cbac ftp timeout 120
ip inspect name cbac tcp
ip inspect name cbac udp
!
ip inspect alert on
!
interface serial 0/0
 ip inspect cbac in
```

4 Assignment 4 – Design Under Fire

For this assignment, I have chosen to attack the project developped by Kevin Olree, whose network diagram is show in the following picture. Kevin practical can be found on the Sans website at http://www.sans.org/y2k/practical/Kevin_Olree_GCFW.doc

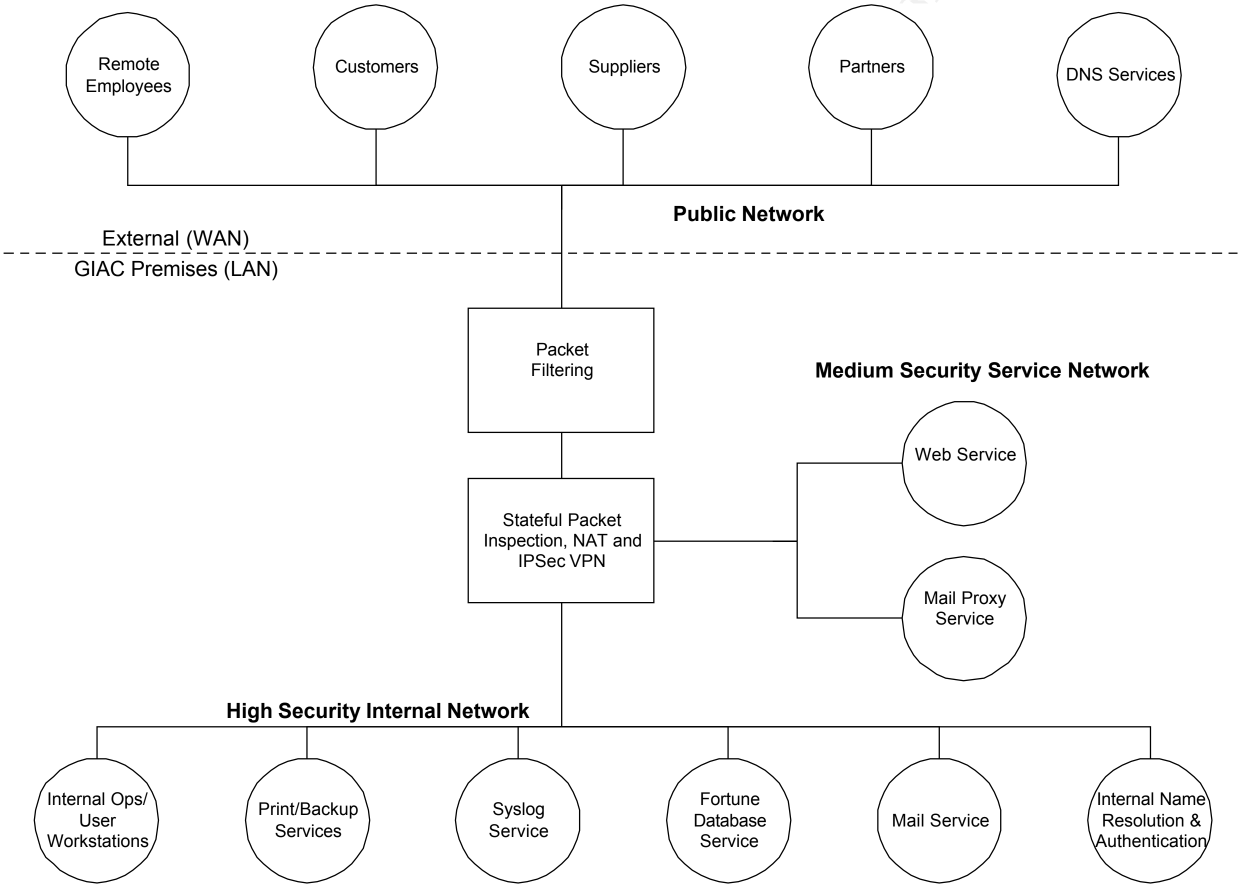


Figure 1.1: Logical Network Design

4.1 *Attacking the firewall*

Kevin uses a PIX 515 (OIX515-UR-BUN) firewall located behind a Cisco 3620 router, that has been hardened to forefront the most evident security issues.

The PIX firewall, and in particular its latest version 5.3, has been rated high in some independent firewall tests. PIX is a Stateful Inspection packet filter and has some extra features like:

1. Mailguard
2. FTP command filtering
3. TCP SYN flood intercept
4. Java and ActiveX filtering
5. FragGuard

4.1.1 PIX Vulnerabilities

Kevin is using PIX software version 5.3. Even if his choise a very good one, the PIX firewall has suffered a series of vulnerabilities, affecting various releases of its software, namely:

PIX and CBAC Fragmentation attack: described in http://www.cisco.com/warp/public/770/nifrag.shtml,
"Because the firewall drops only the initial fragments of blocked datagrams, attackers can exploit this vulnerability by sending streams of complete fragmented packets. The attacker in this case deliberately intends the initial fragments to be blocked by the firewall. Since only the non-initial fragments will be forwarded, the effect on the target host will be similar to the effect of sending only the non-initial fragments to begin with. This method involves some waste of the attacker's resources, and is therefore slightly less effective than simply sending the non-initial fragments alone. This method is of interest because it allows attacks to be launched using relatively standard networking tools, without any special exploit program. ". This problem, however affects PIX version 4.2 or earlier releases.

TCP reset vulnerability: described at http://www.cisco.com/warp/public/707/pixtcpreset-pub.shtml,
“The Cisco Secure PIX Firewall cannot distinguish between a forged TCP Reset (RST) packet and a genuine TCP RST packet. Any TCP/IP connection established through the Cisco Secure PIX Firewall can be terminated by a third party from the untrusted network if the connection can be uniquely determined. This vulnerability is independent of configuration. There is no workaround.”

Cisco PIX Tacacs+ denial of service vulnerability
Bugtraq ID 2551

A problem with the PIX could allow a denial of service. PIX firewalls using TACACS+ are vulnerable to a resource starvation attack which results in a denial of service. Upon receiving multiple requests for TACACS+ authentication from an unauthorized user, the firewalls resources can be exhausted. This causes the firewall to crash, requiring power cycling to resume regular service.

This makes it possible for a user from either the public or private side of the PIX to crash the firewall, and deny service to legitimate users.

All PIX Firewalls (up to PIX version 5.3) having configuration lines beginning with the following line are affected:
pixfirewall# aaa authentication

To exploit this vulnerability, from an internal unix system, execute the following command:

```
while (true); do (wget http://external.system 2>/dev/null &); done
```

Cisco PIX Firewall SMTP Content Filtering Evasion Vulnerability
Bugtraq ID 3365 (formerly ID 1698)

Reference: http://www.securityfocus.com/cgi-bin/vulns-item.pl?section=discussion&id=1698

“Like other firewalls, the Cisco PIX Firewall implements technology that reads the contents of packets passing through it for application-level filtering. In the case of SMTP, it can be configured so only certain smtp commands can be allowed through (for example, dropping extra functionality, such as HELP or commands that could be a security concern, like EXPN or VRFY). When recieving messages, it allows all text through between "data" and "<CR><LF><CR><LF>.<CR><LF>", as this is where the body of the message would normally go and there could be words in it that are smtp commands which shouldn't be filtered. Due to the nature of SMTP and flaws in exceptional condition handling of PIX, it is reportedly possible to evade the smtp command restrictions by tricking the firewall into thinking the body of the message is being sent when it isn't.

During communication with an smtp server, if the "data" command is sent before the more important information is sent, such as "rcpt to", the smtp server will return error 503, saying that rcpt was required. The firewall, however, thinks everything is alright and will let everything through until recieving "<CR><LF><CR><LF>.<CR><LF>". It is then possible for the attacker to do whatever he wishes on the email server.”

All Pix versions are vulnerable.

Here an example of what I could do exploiting this bug:

```
# telnet mail.giac.com 25
<< 220 Giac sendmail version 8.11.1  ready
helo Mail.giac.com
<< 250 mail.giac.com Hello x.y.z, pleased to meet you
MAIL FROM: bill@microsoft.com
DATA ( From here pix disables fixup)
EXPN guest ( Now I could enumerate user
VRFY oracle and have access to all command)
help
whatever command I want
QUIT
```

4.2 Denial of service attack

4.2.1 The router

Even if the external router is accessible only from host 9.9.9.3 for configuration purposes, and no snmp or http access is allowed, some considerations can be made on the ingress filtering proposed by Kevin:

1. The anti spoof filter does not block packets coming from the null address (0.0.0.0)
2. The anti spoof filter does not block packets coming from the loopback address (127.0.0.1)
3. The anti spoof filter does not block packets coming from the null DHCP lease reserved address (169.254.0.0)
4. The anti spoof filter does not block packets coming from broadcast/multicast address
5. A better policy on the ingress filter would require to specify the allowed port/host combinations, then deny all other connections. The list of a limited number of denied ports (69,87,111,2049,512-515, 540, 2000, 6000-6100) and the final permit ip any any in the access-list 101 allows too much traffic to penetrate from the outside through the router.
6. No icmp traffic is screened, with the exception of ip unreachables messages.

Even if some of these considerations does not constitute the evidence of vulnerability to be exploited, they can be used to hide the identity of the attacker, to enable a DOS attack against the router itself and to allow the penetration of undesired traffic towards the firewall.

Furthermore, ICMP is one of the most exploited protocols, being used for reconnaissance, denial of service attacks, and more.

The routing table of the router can be subverted using icmp redirect messages, rendering the GIAC network isolated from the rest of Internet, or ICMP unreachable packets can be crafted to block GIAC the access to partner/suppliers sites, or to the sites from where GIAC downloades updates to virus signatures, patches to applications, operating systems, and so on.

A redirect ICMP packet can also be used to subvert the router security causing traffic to flow via a path the network manager didn't intend.

4.2.2 Distributed DOS

These considerations are also valid if we plan a distributed denial of service. The effects are enormously amplified in this case, because DDOS tools are designed to disrupt normal site operations by flooding the network with a large amount of traffic. Most modern DDOS tools use a multi tier architecture.

Typically the hacker uses a telnet connection toward a number of masters (compromised systems) to control a large population of daemons that actually generate the DOS traffic against the victim.

We will utilize Trinoo, and for our purposes, we will utilize two compromised system as masters. Each Trinoo master will control 25 Trinoo daemons that will overwhelm the Kevin network from DSL connections to the Internet. Attack targets can be the external router, the external DNS server, the Web server or the Mail server of Kevin. Trinoo daemons flood the victim network with UDP packets, and UDP is also used get control of the daemons by the masters.

The following communication channels are used by Trinoo:
1. intruder -> master : destination port tcp/27665
2. master -> daemons : destination port udp/27444
3. daemons -> master : destination port udp/31335

4. daemons -> victims : UDP flood with random destination ports

For more information on Trinoo, TFN and other DDOS attacks/tools, see the following docs:
1. http://www.cert.org/incidents_notes/IN-99-07.html
2. http://www.cert.org/advisories/CA-2000-21.html
3. http://xforce.iss.net/alerts/advise40.php
4. http://razor.bindview.com/publish/advisories/adv_NAPTHA.html
5. http://staff.washington.edu/dittrich/misc/trinoo.analysis

Defending against any DDOS attack is extremely difficult. Bandwidth starvation is easily accomplished and the filtering capabilities of the router or the firewall are rapidly saturated. Things are further complicated by the fact that the daemons are usually not under the same administrative domain, so many companies and/or ISPs are to be contacted before the attack can be stopped.

The only serious countermeasure that can be put in place is to avoid that any company network be used to attack somebody else. This can be obtained implementing the policy rule of being a good Internet neighbor and implementing egress filtering on the border router. There is nothing a site can do with currently available technologies to prevent becoming a victim of coordinated network flood.

A good paper that details a strategy for defeating distributed attacks, authored by Simple Nomad, is available at http://razor.bindview.com/publish/papers/strategies.html.

The "Results of the Distributed-Systems Intruder Tools Workshop", available at http://www.cert.org/reports/dsit_workshop-final.html suggest some recommended actions for coping with the potential for an attack using distributed system intruder tools, among the others:

1. Become fully informed
2. Be cognizant of your own site's security posture
3. Assess the services that are mission critical for your particular business
4. Develop an augmentation strategy to provide staff and other resources in the event of an attack
5. Be sure the staff have the time and resources needed to perform traffic analysis, intrusion detection, coordination with upstream providers
6. Ensure privacy issues in log retention are properly addressed
7. Ensure responsibility is correctly addressed in the current policy requirements
8. Be sure that all levels of management understand and are held accountable for security planning and implementation
9. Define security resources in the budget
10. Develop cooperative relationships with other sites (es: ISP, incidents response organizations, etc.)
11. Pressure vendors to provide more security in their products
12. Apply anti spoofing rules at the network boundary
13. Enable detection of unsolicited ICMP echo replies and unusually high traffic levels
14. Keep systems up to date on patches
15. Follow CERT/CC and SANS best practices
16. Establish reference systems using cryptographic checksum tools
17. Periodically compare systems to the reference copy
18. Run host based software to detect vulnerabilities and intrusions

19. Scan the systems periodically looking for well know vulnerabilities and correct the problems
20. Deploy an IDS
21. Provide security training for users
22. Create and practice a response plan
23. Establish detailed written plans for communicating with IRTs, ISPs, and law enforcement agencies
24. Develop a forensic toolkit to assist in forensic analysis

4.3 Compromising an internal system

In accomplishing this task I select, as a target of my attack, the web server of the Kevin Network. I choose this system for several reasons:

1. It is one of the exposed servers in the service network
2. I assume it based on Microsoft IIS 5 on top of a Windows 2000 server
3. IIS is probably the most attractive target to hackers in these days
4. IIS has a series of buffer overflows/vulnerabilities
5. New vulnerabilities of IIS are revealed periodically, and perhaps Kevin has not yet patched the server and/or a countermeasure is not yet available

A CERT advisory at http://www.cert.org/advisories/CA-2001-10.html reveals that Windows 2000 includes an ISAPI extension providing support for the Internet Printing Protocol. The ISAPI/IPP extension contains a buffer overflow that can be used by an attacker to execute arbitrary code in the Local System security context. Complete control of the system can thus be gained.

To protect against this vulnerability, Microsoft has issued a patch. More information is available at the Microsoft site http://www.microsoft.com/technet/security/bulletin/MS01-023.asp

Starting from the end of July 2001, a new series of attacks is propagating on the Internet, due to an increasing number of systems compromised by a new tool called Red Code.

Red Code is a IIS 4.0/5.0 Worm that spreads through TCP port 80 (HTTP). It scans random IP addresses to infect other systems. It is code residing in memory only, so no files are injected on the infected systems. A good resume of Red Code has been written by Jason Fossen of the SANS Institute.

Red Code affects the following systems:

1. IIS 4.0 and IIS 5.0
2. Windows NT 4.0 with Option Pack.
3. Windows 2000 Server and Advanced Server installs IIS by default.
4. Cisco 600 Series DSL Routers.
5. Other HTTP-enabled devices being adversely affected too.

Red Code is a Buffer overflow in IDQ.DLL, the ISAPI Extension for .ida and .idq files. These are files used by Indexing Service, but this service does not need to be running. IDQ.DLL runs in Inetinfo.exe by default, which runs as Local System.

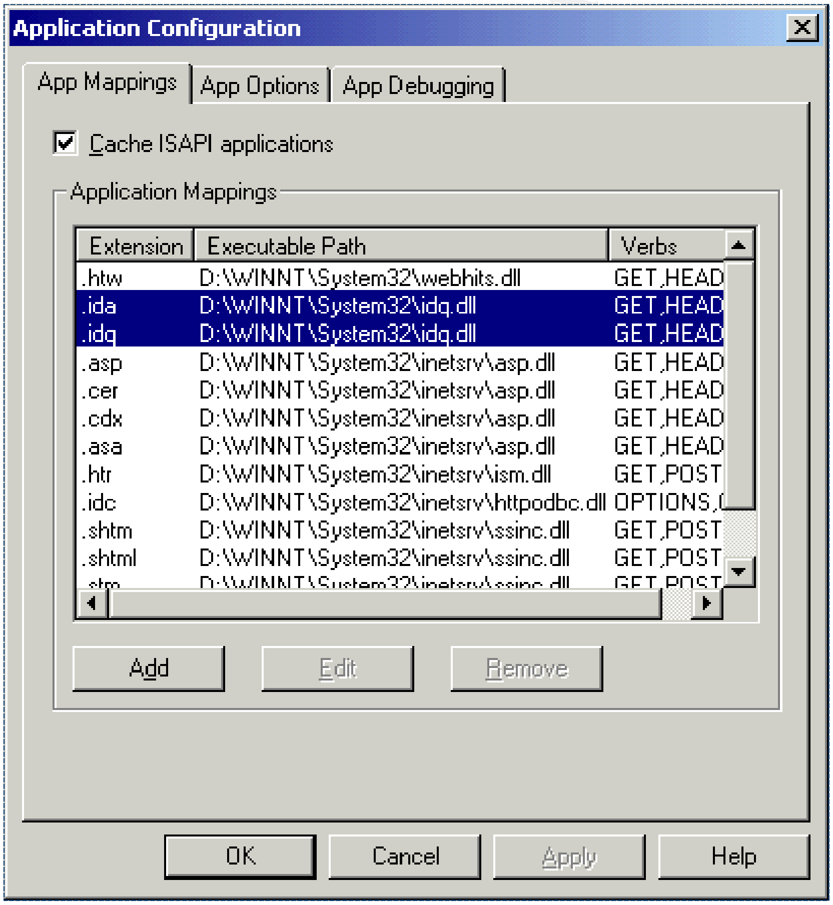
Injected code is embedded in the initial GET request. The following is a typical attack pattern logged:

```
GET /default.ida NNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNN%u9090%u6858%ucbd3%u7801%u9090%u6858%ucbd3%u7801%u9090%u6858%ucbd3%u7801%u9090%u9090%u8
190%u00c3%u0003%u8b00%u531b%u53ff%u0078%u0000%u00=a
```

Microsoft has posted a security bulleting regarding the countermeasurs/patches to protect against this attack at http://www.microsoft.com/technet/security/bulletin/MS01-033.asp

But it is also possible to stop new ISAPI Extension buffer overflow attacks before hackers even discover them using the following procedure:

1. In the "Internet Services Manager" tool, right-click on each website, and select Properties.
2. Click on the "Home Directory" tab.
3. Click on the Configuration button.
4. Click the "App Mappings" tab.
5. These are your ISAPI Extensions!
6. Delete the associations you want to disable (in the following figure .ida and .idq)



☑ Cache ISAPI applications

Application Mappings

Extension	Executable Path	Verbs
.htw	D:\WINNT\System32\webhits.dll	GET,HEAD
.ida	D:\WINNT\System32\idq.dll	GET,HEAD
.idq	D:\WINNT\System32\idq.dll	GET,HEAD
.asp	D:\WINNT\System32\inetsrv\asp.dll	GET,HEAD
.cer	D:\WINNT\System32\inetsrv\asp.dll	GET,HEAD
.cdx	D:\WINNT\System32\inetsrv\asp.dll	GET,HEAD
.asa	D:\WINNT\System32\inetsrv\asp.dll	GET,HEAD
.htr	D:\WINNT\System32\inetsrv\ism.dll	GET,POST
.idc	D:\WINNT\System32\inetsrv\httpodbc.dll	OPTIONS,(
.shtm	D:\WINNT\System32\inetsrv\ssinc.dll	GET,POST
.shtml	D:\WINNT\System32\inetsrv\ssinc.dll	GET,POST
.stm	D:\WINNT\System32\inetsrv\ssinc.dll	GET,POST

The following is a series of pointers regarding Red Code:

- **Original eEye Digital Security Analysis of Code Red:** http://www.eEye.com/html/Research/Advisories/
- **eEye Code Red Scanner Tool:** http://www.eEye.com/html/Research/Tools/
- **CERT Advisory CA-2001-19 on Code Red:** http://www.cert.org/advisories/CA-2001-19.html
- **Microsoft Code Red patch:** http://www.microsoft.com/technet/security/bulletin/MS01-033.asp
- **Cisco 600 DSL Router patch:** http://www.cisco.com/warp/public/707/cisco-code-red-worm-pub.shtml

5 References

5.1 References

Cheswick,W. and Bellovin S., “Firewalls and Internet Security”, Addison-Wesley,
ISBN 0-201-63357-4

Chapman, D. B. and Zwicky E., “Building Internet Firewalls”, O’Reilly and Associates,
ISBN 1-56592-124-0

Garfinkel S. and Spafford G., “Practical unix and Internet security”, O’Reilly and Associates,
ISBN 1-56592-148-8

Stallings W., “Cryptography and Network security”, Prentice Hall,
ISBN 0-13-869017-0

McClure S., Scambray J., Kurtz G., “Hacking Exposed”, Osborne/McGraw-Hill
ISBN 88-7303-643-0 (italian edition)

Anonymous, “Maximum Linux security”, Macmillan Computer publishing
ISBN 88-7303-616-3 (italian edition)

Russel R., et al. “Hack Proofing your network. Internet Tradecraft”, Syngress Publishing Inc.
ISBN 1-928994-15-6

5.2 Web References

http://advice.networkice.com/advice/
http://cve.mitre.org
http://doc.cds.unina.it/dux/DOCS/HTML/MAN/MAN8/0293____.HTM
http://www.eEye.com/html/Research/Advisories/
http://www.eEye.com/html/Research/Tools/
http://pasadena.net/cisco/secure.html
http://razor.bindview.com/publish/advisories/adv_NAPTHA.html
http://razor.bindview.com/publish/papers/strategies.html
http://staff.washington.edu/dittrich/misc/trinoo.analysis
http://www.blighty.com/products/spade/
http://www.cert.org/
http://www.cert.org/advisories/CA-2000-21.html
http://www.cert.org/advisories/CA-2001-10.html
http://www.cert.org/advisories/CA-2001-19.html
http://www.cert.org/incidents_notes/IN-99-07.html

http://www.cert.org/other_sources/websec.html
http://www.cert.org/reports/dsit_workshop-final.html
http://www.cert.org/tech_tips/unix_configuration_guidelines.html
http://www.cert.org/tech_tips/win_configuration_guidelines.html
http://www.cisco.com/univercd/cc/td/doc/cisintwk/ics/cs003.htm
http://www.cisco.com/warp/public/707/3.html
http://www.cisco.com/warp/public/707/4.html
http://www.cisco.com/warp/public/707/advisory.html !!!!!!!!!!!!
http://www.cisco.com/warp/public/707/cisco-code-red-worm-pub.shtml
http://www.cisco.com/warp/public/707/pixtcpreset-pub.shtml
http://www.cisco.com/warp/public/770/land-pub.shtml
http://www.cisco.com/warp/public/770/nifrag.shtml
http://www.cymru.com/~robt/Docs/Articles/secure-bind-template.html
http://www.dns.net/dnsrd/
http://www.enteract.com/~lspitz/audit.html
http://www.enteract.com/~lspitz/linux.html
http://www.enteract.com/~lspitz/nt.html
http://www.es2.net/research/firewalk/.
http://www.insecure.org/nmap/index.html
http://www.insecure.org/nmap/nmap_documentation.html.
http://www.kyuzz.org/antirez/hping/
http://www.microsoft.com/technet/security/current.asp
http://www.microsoft.com/technet/security/email.asp
http://www.microsoft.com/technet/security/web.asp
http://www.microsoft.com/technet/security/bulletin/MS01-023.asp
http://www.microsoft.com/technet/security/bulletin/MS01-033.asp
http://www.pentics.net/denial-of-service/white-papers/smurf.cgi
http://www.psionic.com/papers/dns
http://www.psionic.com/tools/logcheck-1.1.1.tar.gz
http://www.psionic.com/tools/portsentry-1.1.tar.gz
http://www.sans.org/infosecFAQ/securitybasics/host_sec.htm
http://www.sans.org/newlook/publications/ntstep.htm
http://www.securityfocus.com/
http://www.vpn.outer.net/2e/vpn.txt
http://xforce.iss.net
http://xforce.iss.net/alerts/advise40.php