Global Information Assurance Certification Paper

Global Information Assurance Certification
Certified Firewall Analyst
Practical Assignment Version 1.9

GIAC Enterprises

Next Generation Network

featuring Firewalls, Perimeter Protection and VPNs

Kris Vangeneugden
April 2003

TABLE OF CONTENTS

SUMMARY

As part of the SANS GCFW Certification, this paper was written to provide a security architecture with related security policies and tutorials appropriate for a virtual e-business company called GIAC Enterprises.

The whole design is based on requirements retrieved from the company's strategy and business operations described in the beginning of the document.

The network design consists of a filtering Cisco border router, a front-end OpenBSD PF firewall with proxy and IPsec gateway and a back-end firewall for internal segregation.

FOREWORD

No doubt that OpenSource software is getting more and more interesting lately. Today's quality level can often compete with their commercial counterparts.
I started looking at OpenSource firewalls a few years ago, and am surprised by the features that they claim to offer.

This practical was a great opportunity to investigate more in this area and see if we are talking about so called vaporware, or if this software is really capable of what it is supposed to do. I also wondered if those products could indeed be deployed within a company.

While I have exclusively worked with commercial firewalls, I must say that I am impressed by the software that I experimented with for this practical; compared to some expensive commercial firewall implementations, PF on OpenBSD might be a better solution. That is if the operators are familiar with TCP/IP and if you do not need to load balance traffic over the firewall (although this is still possible through content switches). But there is no doubt on the fact that its filtering capabilities are superior to some very expensive commercial implementations.

Note that, since I'm Dutch speaking, you might hit on some weird sentences.

Happy reading,

Kris Vangeneugden

1. SECURITY ARCHITECTURE

1.1 Company Overview

GIAC Enterprises is an e-business that deals in the online sale of fortune cookie sayings. It is currently a small size company of 17 people. The headquarters is located in Belgium, and the remote site in the Netherlands is hosting 2 salesmen. Their business is going very well, as fortune cookie sayings seem to be a booming business. Because the company started with only 3 people there has never been a proper design of its network to support the current growth of employees.

1.2 Company Strategy

The Board of Directors sees a need for a well-designed infrastructure to support their future company. It is expected that the company will grow to 44 people within the coming year. This paper must establish their secure, future network infrastructure that supports this expected expansion.

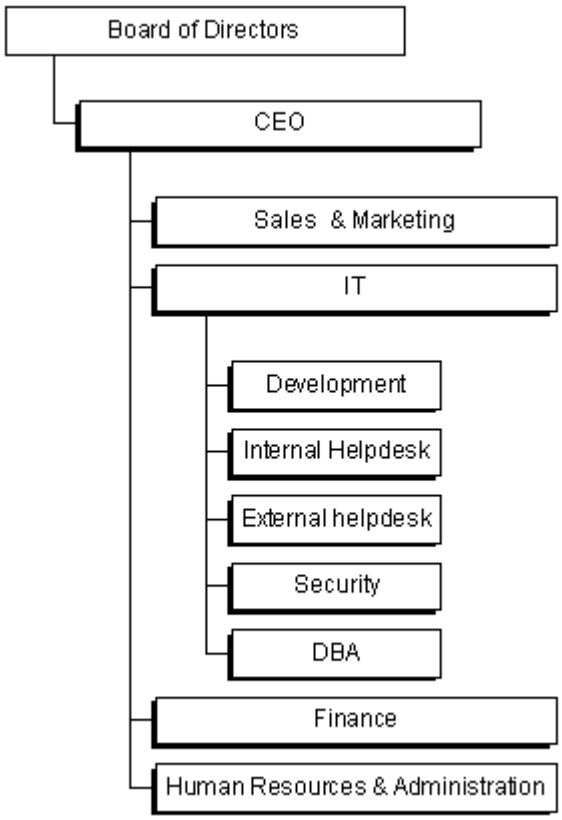
Profit margins within this e-business are rather small, this is why the Board of Directors decided since the founding of the organization that all IT costs must be reduced to the minimum while not endangering the reputation of the company or availability of any of its resources. They understood the power of OpenSource software and the cost savings it would give them. Nevertheless, GIAC Enterprises will financially contribute to the OpenSource communities.

To build their "next generation network", the Board of Directors requests the designer to continue with this mindset. If no OpenSource applications are found that support the organizational or network requirements, commercial products will be selected.

Due to the increased security problems mentioned in the media, the Board of Directors has decided to dedicate two employees to security. Those employees will be responsible for the set-up, maintenance and monitoring of firewalls, intrusion detection systems, anti-virus and all other security devices. They will also need to warn the internal helpdesk incase of security alerts affecting their software.

Although all critical servers must be fault-tolerant, the network path will not be redundant due to budget constraints. The need for network path redundancy will be re-evaluated in 2 years.

1.3 Company Structure



1.4 Business Operations with Requirements

This chapter lists all entities that contribute to or interact with GIAC Enterprises. It describes what those entities do and require from business operational and architectural point of view.

1.4.1 Customers

GIAC Enterprises is aiming at 4000 customers worldwide within 2 years. Customers are usually attracted by the salesmen visiting restaurants.

Fortune cookie sayings must always be retrieved via the GIAC Enterprises' web site. Customers have the possibility to download fortune cookie sayings in several formats, going from a text-file to a MS-Access compatible database file.
Before the actual download, the customer needs to select how many and what kind of sayings he would like to have. The customer is able to view a sample of fortune cookie sayings on the page itself.
Once the selected listing is downloaded, the customer's account is charged based on the number of downloaded sayings.

Invoicing and payments are all done via the website. This website runs on a Apache web server running PHP, and is located at the headquarters.

Access requirements

Visitors access the website through HTTP on port 80.

All customers connect to the GIAC website. Through HTTPS, a valid customer is able to log-in with his username and password received at the online registration. This requires inbound SSL connections on TCP port 443.

Bandwidth requirement

An average of 120 logins per day is expected, with a maximum of 500 on Friday afternoons.

Besides paying customers, GIAC Enterprises expects web site visits from potential customers and regular surfers, resulting in an average of 900 hits per day.

1.5KB/sec for max 20 simultaneous connections (visitors) and 4KB/sec for max 30 simultaneous sessions gives a requirement for around 150KB/sec (including the customers for the partners website that GIAC is hosting).

1.4.2 Suppliers

The suppliers are third parties providing new fortune cookie sayings to GIAC Enterprises.

The supplier is held responsible for the content by strong Service Level Agreements (SLA) to prevent inappropriate fortune cookie sayings being put in our database.

Each record is signed with the suppliers' private key which was generated by GIAC Enterprises generates X.509 certificates using. The text file must conform with the standard message layout (XML is used). When uploaded, the syntax will be checked. If special characters or non-standard fields are discovered, the uploaded file will be rejected. This syntax checking will also prevent the introduction of viruses.

Once the upload finished and is accepted, the file will be stored in the MySQL database.

Access requirements

Suppliers connect through HTTPS (port 443/tcp) to the GIAC website with their username and password. The supplier is then able to upload new fortune cookie sayings to GIAC Enterprises. Their access will not be filtered based on their IP address as several of them use dial-up connections without permanent fixed IP addresses.

Bandwidth requirement

Uploaded files are in average 400KB uncompressed text. An upload via the website is thus ideal. An upload happens once a week and is not time-critical. In a worst case scenario, several partners are uploading a 1MB file simultaneously on Friday afternoon. If the suppliers can upload with 4KB/sec, it would be more than sufficient (less then 2 minutes as it is an uncompressed text file).

1.4.3 Partners

Three international companies translate and resell the fortunes cookie sayings of GIAC Enterprises.

Partners are supposed to login onto the website to download the requested blocks of fortune cookie sayings. Once they download and translate everything, they offer them on their own website.

The latest partner asked GIAC Enterprises to host their translated fortune cookie sayings as they did not want to manage any servers. GIAC Enterprises agreed to give this partner direct access to a dedicated directory on the web server through SFTP. This partner uses a fixed IP address, and only this IP address will be allowed to connect to GIACE with SFTP. This special partner uploads its fortune cookie sayings like a supplier, but will be stored on a separate database instance onto our back-end database server.

This partner is supposed to edit all web pages in their preferred language at their premises and then upload them to the GIAC Enterprises website. Their website will be accessible through http://www.giacenterprisespartner.com which is linked to a dedicated IP address that is redirected to the right page.

Access requirements

Partners login onto the web site within an HTTPS session to download the requested of fortune cookie sayings blocks that they want to translate.

The partner that GIACE is hosting will also be able to upload the translated ones. This is also through HTTPS. They also need to upload their web pages to the web site using to SFTP (source IP restricted).

Bandwidth requirement

Similar to the suppliers.

1.4.4 Employees

All workstations will run Suse Linux 8.2 running KDE 3.1 with OpenOffice 1.0.3 and Mozilla 1.3.

They all have access to the internal file servers, have at least one POP3 account on the internal mail server, and are able to send emails. Each employee is allowed to surf the Internet. The condition is that it should be job related, although some private use is accepted as long as the visited site is appropriate. Everybody can access the GIACE web server. On request, employees are allowed to use passive mode FTP to download files from the Internet.

Each department has however it own specific needs:

- Developers (4 today -> up to 10 within 12 months)
  Most people within the company, 10 people, will be devoted to developing and supporting the web applications of GIAC Enterprises.

  From their workstations they need SFTP and SSH to the web server.

- DBA (2)
  The two Database Administrators are responsible for the correct functioning of the MySQL database.

  From their workstations they need SFTP and SSH to the database server.

- Security (2 within few days)
  These employees will be responsible for the set-up, maintenance and monitoring of

firewalls, intrusion detection system, anti-virus and all other security devices. They also need to warn the internal helpdesk in case of software vulnerabilities.

From their protected LAN, they are able to connect to all servers (syslog, web, database, etc.), firewalls and router. Any server/workstation is able to send syslogs to their server.

- Internal Helpdesk (2)
  Fixing PC problems, installing or upgrading software and hardware for internal and external employees. They can connect to any internal user that has launched "Secure VNC". They also rely on SSH and SFTP. Together with the security personnel, they are the only ones with access to root passwords.

- Customer Support (2 today -> 4 within 5 months)
  These 3 employees will try to answer, by phone or email, all questions that the customer might have when accessing the website.

  They do not have any additional needs.

- Finance, Human Resources and Administration (3 today -> 6 within 12 months)
  This group never deals with customers, nor do they have a clue of the technology GIAC Enterprises deployed. Two of them will work from home, they do not need to connect to the fileserver of Fin/HR/Admin, only the internal file sever (all traffic is tunneled with IPsec ESP using FreeS/WAN on their Suse systems).

  All their PCs and their dedicated file server are on a separate LAN which is firewall protected. They can also access the internal fileserver shared by all other employees.

- Sales (2 today -> 6 within 12 months)
  These sales people spend most of their time at the customer. If not, they are on the phone. In between they send many emails and traditional mailings to GIAC Enterprises' target market.

  They do not have any additional needs.

- Mobile sales force (4 working external within 2 months)
  They do not have any additional needs. But need access to the internal LAN from the Internet. All communication will be tunneled through IPsec ESP with FreeS/WAN running on their Suse systems.

- Remote site (today 2 -> 6 within 12 months)
  Similar to the mobile force, but, have an IPsec gateway at their site.

  The remote office has 24KB/sec upstream speed.

Access Requirements

For the internal users the firewall must be configured to support their connectivity needs as described above.
For all remote users: From any IP address on the Internet port 50/TCP (ESP) and port 500/UDP (IKE) must be allowed. The firewalls must allow connections from the remote user, as described above, to the internal LAN.

Bandwidth requirements

The remote office ADSL line has 12KB/sec upstream connection. File and mail download should not exceed 200MB/day. 90KB/sec should be possible.

1.4.5 Summary of access requirements

Internal users access

Service \ User	Regular User	Developer	DBA	Sec	Internal Helpdesk	Customer Support	HR, etc.	GIACE web site
HTTP(S) to Internet	1	1	1/0	1/0	1	1	1	0
FTP-Passive to Internet	(1)	(1)	1/0	1/0	(1)	(1)	(1)	0
HTTP(S) to web server	1	1	1	1	1	1	1	0
POP3 to mail server	1	1	1	1	1	1	1	0
STMP to mail server	1	1	1	1	1	1	1	0
SCP/SFTP to file server	1	1	1	1	1	1	1	0
SCP/SFTP to HR/F/A file server	0	0	0	1	0	0	1	0
SSH/SFTP to web server	0	1	1	1	1	0	0	0
SSH/SFTP to DB	0	0	1	1	0	0	0	0
MySQL to DB	0	0	1	0	0	0	0	1
S-VNC to all work-stations	0	0	0	0	1	0	0	0
Internal DNS	1	1	1	1	1	1	1	0
ISP DNS	1	1	1/0	1/0	1	1	1	0
Telnet to router	0	0	0	1	0	0	0	0

1: service required
(1): service required for some individuals
1/0: depend on which PC is being used
0: service not required

External access over Internet

Service \ User	Visitor	Customer	Supplier	Partner	Special Partner	External employee	MTA on Internet
HTTP(S) to web server	1	1	1	1	1	0	0
SFTP to web server	0	0	0	0	1	0	0
STMP to mail server	0	0	0	0	0	1	1
POP3 to mail server	0	0	0	0	0	1	0
SCP/SFTP to file server	0	0	0	0	0	1	0
IPsec	0	0	0	0	0	1	0
Internal DNS	0	0	0	0	0	1	0

1: service required
0: service not required

1.4.6 Summary of bandwidth requirements

Access from externally

Maximum bandwidth usage in Kbits/sec:

User / Service	Internal Users	VPN for External employee	Visitor	Customer	Supplier	Partner	Special Partner	E-mail	Total
HTTP(S) to web server	0	8	240	960	64	64	16	0	1352
SFTP to web server	0	0	0	0	0	0	16	0	16
Internet Access	140	0	0	0	0	0	0	0	140
Internal mail and file server	0	72	0	0	0	0	0	60	132
Total	140	80	240	960	64	64	32	60	1640

1.4.7 Summary of availability requirements

All critical servers must be fault-tolerant. They should have dual power supplies, RAID5, doubled memory, etc.

When a machine fails completely, it should be possible to have it replaced within 30 minutes.

Each critical server or network device will be bought in double. Not to use them in a redundant set-up but to have a manual backup solution available incase of problems. When a machine fails completely, it should be possible to have it replaced within 30 minutes. The free PartImage for Linux software will be used to restore entire system partitions quickly. When available, those backup servers and network devices can be used for the qualification of new features without interrupting live operations.

The network path does not have to be redundant due to budget constraints. The need for network path redundancy will be re-evaluated in 2 years.

1.5 Network Requirements and Principles

Although the network must support all existing applications, all access that is not justifiable through the business operations will be blocked by the firewalls.
Firewall filtering must be used to protect all critical servers and to segregate the different security domains.
DoS from Internet may not interrupt internal system resources.
Servers and network equipment have a high MBTF.
Network path redundancy is not required, but, a network component failure may not disrupt the service longer than half an hour.
The design must support network path redundancy, load balancing and network device set-ups in high-availability when required.
Each device must be monitored and managed from a central location.
Server networks must be monitored for protocol anomalies.
Stateful filtering include tcp sequence numbers and icmp

Provide a layered security design, also known as defense-in-depth
While selecting a product, the administrators must be able to use the product.
Be a good Internet neighbor by ensuring that no device at GIAC Enterprises can be used to attack other Internet services.
System events are stored on the local system and collected and stored on a central system
SSH and SFTP are used for management of systems
Due to few and static records, DNS records are maintained and controlled by the ISP.
Design should be able to cope with the 130% expected growth.
Maintenance period of 3 hours (from 5 till 8am) on the 1st and 3rd Sunday of the month.
From the internal server LANs, Internet access and email are not available

1.6 Network Design

Based on the company structure, the business operations requirements, the strategy and network security requirements, the following design is proposed.

1.6.1 Physical design

The key-elements contributing to the security of the security of GIAC Enterprises' network are:

Dual firewall setup based on OpenBSD 3.2 with PF firewall.
A front-end firewall coping with Internet originated attacks directed at the Internet offered services, and a back-end firewall controlling back-end and back-office connectivity

Filtering on the Cisco 1760 Internet router

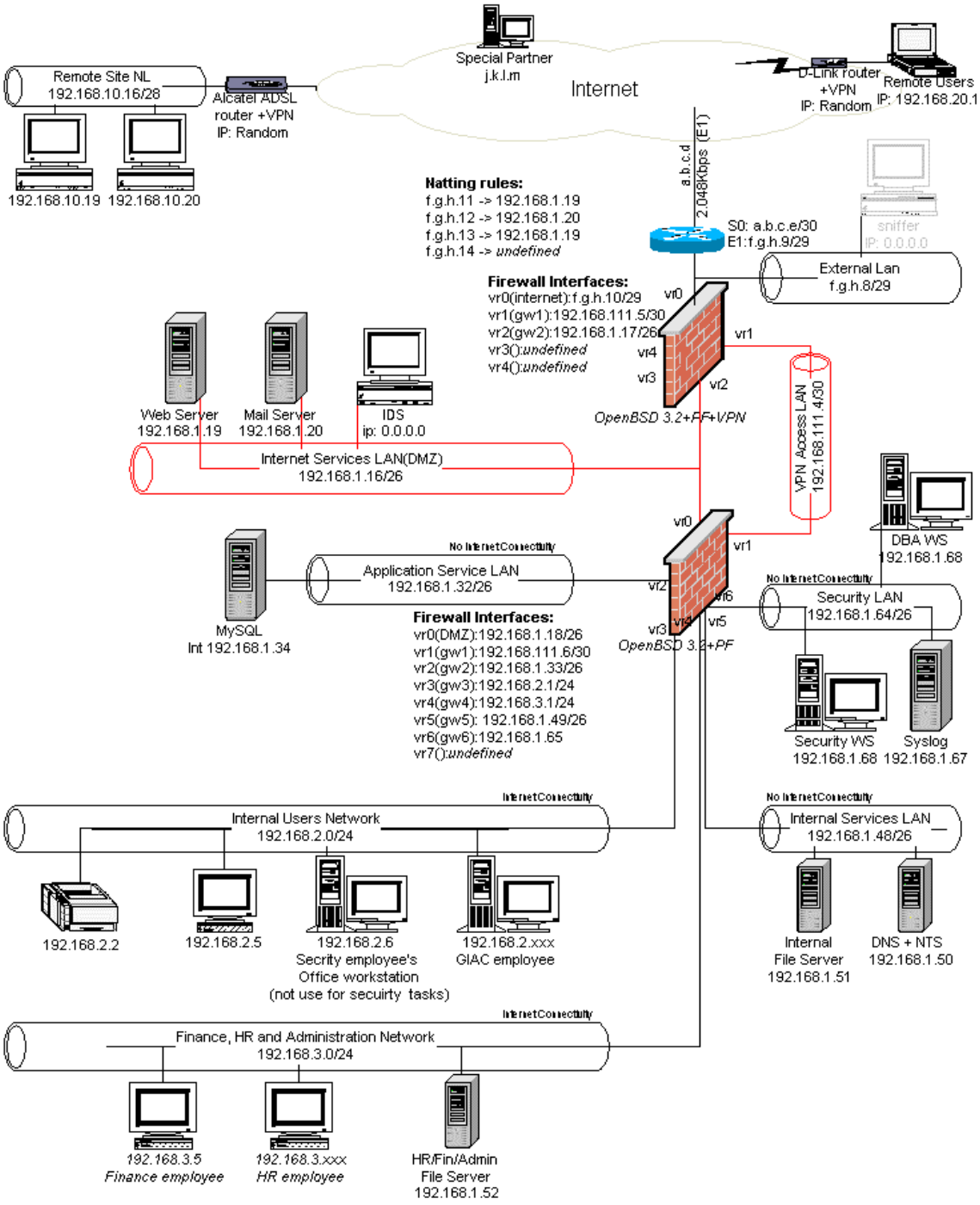
2.048Mbps E1 leased line to the ISP

A tiered architecture with network segmentation: Servers directly accessed by the Internet community will only act as a front-end located into a DMZ. The actual data is securely stored in a back-end server located in a highly access restricted network

A Debian GNU/Linux, Woody release, that is Bastille hardened as operating systems for all servers with latest security patches applied.

Host-based intrusion detection with Tripwire on the Internet Services LAN (DMZ) and Application LAN.

Workstations that have root to critical servers are located in the security LAN. This LAN has no Internet access.



1.6.2 Filtering border router

Product

A Cisco 1760 router with 96MB of memory running Cisco IOS release 12.2(16) without additional software features. Cisco was selected because the network manager is already familiar with IOS software.

The 2 WIC slots are filled with:
- WIC-1T One-port serial, asynchronous and synchronous WAN Interface Card
- WIC-1B-S/T One-port ISDN BRI (S/T interface)

Function

The router's primary function is to route traffic between the GIAC Enterprises network (interface E0) and Internet (Interface S0).

The synchronous serial supports data rates up to 2.048Mbps for the E1 leased line to the ISP. The ISDN port (bri0) will be used in case of a failure of the E1 link.

The router's secondary function is to sanitize the entering IP packets (first line of defense):
- apply anti-spoofing rules (ingress and egress filtering)
- prevents private IP packets leaking through the firewall.
- block internal broadcast addresses preventing SMURF amplifiers
- disallow source-routed packets

This is done through static packet filtering (by defining ACLs) and some IOS commands.

The router also provides the correct time to the internal network through authenticated NTP. The router itself is synchronized with the ISP NTS.

1.6.3 Front-end firewall + VPN concentrator + Proxy

Product Description

OpenBSD 3.2 will be installed on a Dell PowerEdge 2650, with a quad-ethernet PCI NIC and a PowerCrypt encryption accelerator PCI card.

OpenBSD provides strong security. Only one remote hole in the default install, in more than 7 years. This operating system can be considered as the number one in the industry for security.

The OpenBSD Packet Filter (OpenBSD PF) is a real stateful firewall that is part of the kernel since OpenBSD 3.0. PF includes also ALTQ. This provides resource-sharing and Quality of Service for network traffic.
PF is much easier to configure than iptables. Furthermore, due to its all-in-one-file approach, it is even easier than a big Check Point Firewall-1 through all its GUIs and hidden settings.

PF it is entirely command-line base. Fortunately, there is no need for nice GUIs because there are two well-trained security people familiar with BSD.
Note that a graphical GUI, called fwbuilder, can be used if the rulebase becomes too complex.

The scripting functionality of PF enables you to write your own scripts that can be triggered by e.g. an IDS.

OpenBSD comes with an IPsec stack that is enabled by default.

Squid version 2.5, an open-source package, will be additionally installed. Squid is a high-performance proxy caching server for HTTP data objects, but also supports FTP. Squid handles all requests in a single, non-blocking and I/O-driven process. Squid supports SSL, extensive access controls, and full request logging. OpenBSD 3.2 offers the squid-2.5.PRE13.tgz packages. So, we use a port for Squid2.5-Stable1 available on the OpenBSD port collection.

Function

The front-end firewall is supposed to protect the internal network from unauthorized access to internal services and attacks originating from the Internet. It does this by inspecting packets traveling between the internal network and Internet, and imposing restraints upon the sources, targets, session types and even content of these packets.

Its second function is act as a VPN concentrator for all remote employees, remote site(s) and partner(s). All IPsec tunnels will be terminated on this firewall.

Its third function is to proxy all HTTP data objects and to terminate SSL connections to the web server. Squid will be used for this purpose.
Squid will mainly be used to:

- protect the Apache web server by performing access control and filtering
- free some resources of the web server
- to terminate the SSL connection so that IDS in the "Internet service network" can inspect the traffic.
- speedup web access for internal users.
- block images from ad-servers for internal users.
- block sites that should not be visited by internal users.
- cache DNS lookups

The encryption accelerator card provides additional system resources for the cryptographic calculations required for IPsec and SSL.

OenBSD PF does not support a High Availability setup, but, when HA is required, the design supports the addition of content switches in front of 2 OpenBSD PF firewalls.

1.6.4 Back-end firewall

Product Description

Similar to the front-end firewall hardware. OpenBSD 3.2 on a Dell PowerEdge 2650, but with two quad-ethernet PCI NIC and no encryption accelerator card and IPsec disabled.

Function

The back-end firewall is used to segregate the internal networks by inspecting packets traveling between these networks, and imposing restraints upon the sources, targets, protocols and even content of these packets.

Due to a back-end firewall in addition to a front-end firewall, the internal communications will not be affected when the front-end firewall gets flooded. Secondly, a configuration error in the front-end firewall (rulebase or IPsec related) will not directly endanger GIAC Enterprises' core data stored on the internal servers.

1.6.5 Network intrusion detection

Product Description

Snort v2.0 on Woody (Debian GNU/Linux latest stable release) running on a Dell PowerEdge 2650 with a quad-ethernet. This host will be dual homed.

Snort was one of the screened products because it has a strong HTTP flow analyzer, does protocol anomaly detection (IP, TCP, UDP, ICMP, RPC, HTTP, etc), does stateful-pattern matching, and can handle IP fragmentation.
The preference went to Snort because it is free and the security team is knowledgeable enough to deploy and maintain it.

Function

In case of an attack, the firewall will only create few log entries showing that their were some hits on port x. Without looking at the packet itself, the firewall log analyzer will not know if the packet itself was “evil” or not.
With network IDS, GIACE will know if an evil IP packet was sent to one of their services.

GIACE does not want to install a network IDS in front of the Internet firewall (that is between Internet router and front-end firewall). They do not have the intention to back-trace someone that tried a SMB exploit for Windows XP to one of their servers. This enables them to focus on the anomalies in the allowed traffic.

When the IDS operator confirms an attack to one of the servers, the firewall will configured to block the originating IP address.

1.6.6 Web server

Product Description

The web server is a Dell PowerEdge 2650, running Apache version 1.3.27 on the latest stable Debian GNU/Linux release (Woody).

The freely available Apache HTTP sever is the most popular web server on the Internet since April 1996.

Apache will be configured inline with security tips provided by the Apache team on http://httpd.apache.org/docs-2.0/misc/security_tips.html .

Function

The web server will represent GIAC Enterprises on the Internet. It enables them to do business.

Besides a strong perimeter protection, GIAC Enterprises requested to apply defense-in-depth. As web servers are often a weak point, running a locked down Apache on a secured Debian GNU/Linux Woody system, will definitely contribute to a defense-in-depth architecture.

1.6.7 DNS

GIAC Enterprises will not run a DNS server as only 4 static records are needed. The ISP will take care of our DNS records.

A small internal DNS is deployed for internal name resolution. For Internet users, the ISP DNS is used. Note that Squid caches DNS lookups.

1.6.8 Security domains

At GAIC Enterprises, different security domains are segregated by firewalls. Each security domain is enforced through a LAN. The security domains differ from each other due to their population, usage and criticality.

Note that by having critical servers in a separate LAN, remote exploits can be better controlled (few services opened though the firewall), and (*) when Internet access is blocked for that LAN, no information can leak out directly to the Internet (as this connectivity is missing).

The security domains are also supported by a physical security infrastructure.

Seven security domains are defined:

Internet Services LAN
This is the network that contains the least trusted systems, being accessed by the Internet community. Although those servers are time-critical, GIAC Enterprises would not suffer from business loss if a hacker deletes the whole content of this server.

Application Services LAN
This network contains the companies' crown jewels. If this machine gets hacked, the business might be heavily impacted. It contains all data that is being sold; the customers database, etc. (*)

VPN Access LAN
This is not really a LAN. This LAN does not contain any hosts, but has been created to connect the front-end firewall with the back-end firewall. This prevents hackers in the DMZ to masquerade a remote user's IP address. (Alternative was to build a service-leg DMZ, but that would provide lower performance and less security).
Everybody with a valid certificate can VPN to the VPN concentrator, to access some internal hosts.

Internal Services LAN
This network hosts some servers that are commonly used by the GIACE employees. These servers are critical and contain information that could be interesting for some people. (*)

Security LAN
The security servers and workstations must be strongly protected, because from these hosts, all devices on the network can be accessed with highest privileges. Furthermore, all the security is enforced and controlled from these workstations. All security events are collected on the syslog server. It allows the security administrators to grep and correlate logfiles on one server (rather than obtaining the information from multiple logfiles on multiple devices). In addition, storing all logfiles on a remote logging server makes it harder for a network intruder to cover his tracks. (*)

Internal User Network
On this network, the actual work is being done. It hosts equipment used by the developers, the helpdesk, the salesmen, etc. There are no risks in putting those groups on the same network.

HR, Fin & Admin Network
This network contains PCs dealing with salaries, sensitive employee info, etc. This data must be protected against some internal threats.

1.6.9 Network addressing

External Addresses

GIAC Enterprises would not use more than six (non-natted) public IP addresses. Therefore, the ISP assigned f.g.h.9/29 to us. So without even using port redirecting, we can easily setup 4 different servers (firewall and router take the 2 other IPs).

IP address	Device	DNS
f.g.h.9	Router	/
f.g.h.10	Firewall + VPN	gw.giacenterprises.com
f.g.h.11	Web server	www.giacenterprises.com
f.g.h.12	Mail server	mail.giacenterprises.com
f.g.h.13	Web server partner	www.giacenterprisespartner.com
f.g.h.14	- Available -	/

Internal addressing scheme

Network addressing scheme must take the expansion in to account. The company network will use non-routable addresses.

For efficiency reasons and ease of administration the network (including remote offices) is split up into several class-C networks. Those networks are subnetted into 26 or 28-bit mask subnets with respectively 62 and 14 hosts.
Each subnet represents a physical location or a group of machines. There was no need to match a subnet with each organizational unit.

Internally, the headquarters will use the private C-classes:

192.168.1.0 for all server LANs
This /24 will be subnetted into 28-bit mask subnets

- Internet Services LAN – 192.168.1.16/28
- Application Services LAN – 192.168.1.32/28
- Internal Services LAN – 192.168.1.48/28
- Security LAN – 192.168.1.64/28

192.168.2.0 for workstation LAN 1 (Internal Users Network)
192.168.3.0 for workstation LAN 2 (HR, Finance & Admin Network)
That is not efficient as will never be filled up with more than 50 hosts, but, a /27 would not provide enough hosts, a /26 not enough subnets.

All (future small) remote sites will use C-class
192.168.10.0
This will be subnetted into 14 28-bit mask subnets (supporting 14 hosts):

- Remote site NL – 192.168.10.16/28
- Remote site 2 – 192.168.10.32/28
- Remote site 3 – 192.168.10.48/28
- …

All (future) remote users will get an IP address from the C-class
192.168.20.0, starting with 192.168.20.1. Those IPs are individually provisioned into the firewall.

IP addresses assignment

- The first IP address in the subnet will be reserved for that network's gateway.
- The following IP(s) will be assigned to printer(s)
- The other can be used by servers, workstations or other network equipment

Network Address Translation (NAT)

- Hide NAT
  All Internet destined IP packets, coming from any GIACE workstation, will originate from the firewalls' IP address once the packet passes the firewall. All GIACE initiated Internet traffic will have f.g.h.10 as source address.

- Static NAT
  When someone on the Internet sends a request to f.g.h.11, the firewall will relay it to 192.168.1.19, f.g.h.12 to 192.168.1.20 and f.g.h.13 to 192.168.1.19

1.6.10 Hostname convention

Each IP address will have unique hostname associated with it. For ease of use and identification a naming convention is required as from the start of the design.
Names will all be 9 characters long. The first 2 characters represent the location, the 3+4 represent to type of host, 5-7 represent the department or service type, 8+9 is number to make the name unique.

The different equipment will be identified as:
- PCs: pc
- Laptops: lt
- Printers: pr
- Servers: sv
- Routers: rt
- Switches: sw
- Firewalls: fw

Departments and services will be identified as:
- HR: hrs
- Purchasing: pur
- Finance: fin
- Web: www
- Network management: nwm
- Firewall management: fwm

Examples: "brpchrs03" is a pc from an HR employee located in the Brussels headquarters. "brsvfwm01" is the firewall management server located in Brussels.

2. SECURITY POLICY AND TUTORIAL

This chapter shows how the policies configured for the border router, the OpenBSD front-end firewall and the VPN concentrator. Installation and configuration procedures of the Primary firewall will be explained in detail.

As by GIACs corporate security policy, all communications between the different network security zones must be denied by default. The allowed connectivity must be justified by the business.

2.1 Border Router

Besides configuring the interfaces of the router, some configuration needs to be done to:
- make the system itself more secure by armoring the router
- make the system only accessible from the right location
- provide some protection to the company's network

The router will not be configured to filter traffic, directed to the GIAC's public IP range f.g.h.8 0.0.0.7, on ports. Because:
- this would prevent PF from capturing portscans
- the router does not filter statefully

2.1.1 Armoring the router

A service that is not listening can not be exploited nor provide information to hackers. That is why it is preferred to disable all services that GIAC Enterprises will not require.

Following services will not be used…

> echo, discard, chargen daytime, Cisco Discovery Protocol, finger, bootp, web configuration, SNMP, etc. There is no need to e.g. telnet hostname, thus domain lookups can be disabled.

… and this can be disabled with the following commands:

```
no service tcp-small-servers
no service udp-small-servers
no cdp
no service finger
no ip http
no ip bootp
no snmp
no ip domain-lookup
```

2.1.2 Limit router access

Physical access

Although the router should be located in a physically protected room, it would not hurt if the unconnected console and aux port are non-default password protected (even though these can be reset easily).

```
line console 0
password **************

line aux 0
password **************
```

Network access

A password and ACL filtering on source IP will limit the hosts that can access the device through telnet (on vty 0 to 4):

```
line vty 0 4
access-class 100 in
password 7 ***************
login
```

```
access-list 100 permit tcp IP-NW-Management1 any eq 23 log
access-list 100 permit tcp IP-NW-Management2 any eq 23 log
```

Each access attempt will be logged.

Make sure that nobody will be able to read router passwords from the screen of configuration print-outs:

```
service password encryption
```

Use non-reversible encryption (MD5) to protect the enable password:

```
enable secret
```

2.1.3 Protect the corporate network

Prevent:

- potential harmful packets from reaching destinations that should not be accessible due to an access list (a)
- malicious directed broadcasts from causing denial of service (a)
- sending out network information through ICMP error messages (a)
- traffic originated from unusual addresses entering the network (b)
- spoofing attacks (b)
- private addresses leaking out (NAT-ing error?) (c)

by (a) issuing following IOS commands:

```
! prevent packet filters be fooled by all processing specially routed packets
no ip source-route
! to protect against smurf attacks etc.
no ip direct-broadcast
! prevent network mapping through generating ICMP messages
no ip unreachables
```

by (b) defining access control lists inbound on the serial interface (in global configuration mode):

```
interface serial0
ip address a.b.c.e 255.255.255.252
```

```
ip access-group 101 in
```

```
! Ingress filtering
! Deny incoming RFC1918 addresses – and log this when it happens
access-list 101 deny ip 192.168.0.0 0.0.255.255 any log
access-list 101 deny ip 172.16.0.0 0.15.255.255 any log
access-list 101 deny ip 10.0.0.0 0.255.255.255 any log

! Deny incoming packets with localhost, broadcast, multicast, no or
! GIAC Enterprises ip addresses – log this when it happens
access-list 101 deny ip 127.0.0.0 0.255.255.255 any log
access-list 101 deny ip 255.0.0.0 0.255.255.255 any log
access-list 101 deny ip 255.0.0.0 0.255.255.255 any log
access-list 101 deny ip 224.0.0.0 7.255.255.255 any log
access-list 101 deny ip host 0.0.0.0 any log
access-list 101 deny ip f.g.h.8 0.0.0.7 any log

! Only allow packets with expected destination IP addresses to pass
! the router (Note: Internet can not talk to routers e0 IP address)
! Source addresses are sanitized by rules above.
! The web server
access-list 101 permit tcp any host f.g.h.11 eq 80
access-list 101 permit tcp any host f.g.h.11 eq 443
access-list 101 permit ip any host f.g.h.11
! Mail server
access-list 101 permit tcp any host f.g.h.12 eq 25
access-list 101 permit ip any host f.g.h.12
! Special partner web server
access-list 101 permit tcp any host f.g.h.13 eq 80
access-list 101 permit tcp any host f.g.h.13 eq 443
access-list 101 permit ip any host f.g.h.13
! Available address
access-list 101 permit ip any host f.g.h.14
! For the VPN concentrator, and all returning traffic from
! internal users surfing the Internet:
access-list 101 permit ip any host f.g.h.10

! Allow only specific ICMP error-messages to enter the network:
! net-unreachable, host-unreachable, port-unreachable, packet-too-big,
!  source-quench, administratively-prohibited and ttl-exceeded.
! These rules are put after the “ip permit line” as that one will have most
! matches. Such ordering might provide better performance.
access-list 101 permit icmp any f.g.h.8 0.0.0.7 3 0
access-list 101 permit icmp any f.g.h.8 0.0.0.7 3 1
access-list 101 permit icmp any f.g.h.8 0.0.0.7 3 3
access-list 101 permit icmp any f.g.h.8 0.0.0.7 3 4
access-list 101 permit icmp any f.g.h.8 0.0.0.7 3 13
access-list 101 permit icmp any f.g.h.8 0.0.0.7 4
access-list 101 permit icmp any f.g.h.8 0.0.0.7 11 0
! Allow echo replies coming back – is stateful filtered by firewall
access-list 101 permit icmp any host f.g.h.10 echo-reply
access-list 101 deny icmp any any log

! All valid connections have been described in previous rules.
! Therefore, all other traffic must be dropped:
access-list 101 deny udp any any log
```

```
access-list 101 deny tcp any any log
access-list 101 deny ip any any log
```

Note: Defining the 4 rules above instead of "access-list 101 permit ip any f.g.h.8 0.0.0.7", enables GIACE to see some nice statistics.
Executing "show ip access-list" on the router will list each rule with the number of times that rule was matched. It gives a quick overview of which protocols/host are mostly used. If, for example, most hits are on "access-list 101 permit udp any host f.g.h.11" instead of "… eq 80" the operator knows that something is wrong.

by (c) defining access control lists inbound on the ethernet interface:

```
interface ethernet0
ip address f.g.h.9 255.255.255.248
ip access-group 102 in
```

```
! Egress filtering
! Deny outgoing RFC1918 addresses
access-list 102 deny ip 192.168.0.0 0.0.255.255 any log
access-list 102 deny ip 172.16.0.0 0.15.255.255 any log
access-list 102 deny ip 10.0.0.0 0.255.255.255 any log

! Allow tcp/udp packets from GIAC Enterprises' network to go to Internet.
access-list 102 permit ip host f.g.h.11 any
access-list 102 permit ip host f.g.h.12 any
access-list 102 permit ip host f.g.h.13 any
access-list 102 permit ip host f.g.h.14 any

! Allow internal users to send an echo-request
access-list 102 permit icmp host f.g.h.10 any echo-request

! Drop and log all other traffic
access-list 102 deny icmp any any log
access-list 102 deny udp any any log
access-list 102 deny tcp any any log
access-list 102 deny ip any any log
```

Note: same as (b).

2.1.4 Other security measures

By showing a router login banner, the company is legally better protected in case the system got accessed and compromised by someone.

```
banner / UNAUTHORIZED ACCESS TO THIS NETWORK DEVICE IS PROHIBITED.
You must have explicit permission to access or configure this device. All activities
performed on this device may be logged, and violations of this policy may result in
disciplinary action, and may be reported to law enforcement. There is no right to privacy
on this device./
```

All logs should be send to the syslog server:

- hacker can not erase tracks

- all logs can be analyses from one central server

```
! provide the ip address of the syslog server
logging 192.168.1.67
! get sufficient information on the syslog server
logging trap notification
! limit the logging messages displayed on the console terminal
logging console emergencies
```

The border router will be synchronized with our ISP NTP server. This enables correlation of events collected from the different devices:

First set the time manually:

```
clock set hh:mm:ss month day year
! enables time stamps on logging messages,
! showing the current time and date relative to the local time zone service timestamps
debug datetime localtime
service timestamps log datetime localtime
clock timezone GMT+1
clock summer-time zone recurring
```

Enable synchronization:

```
! All ntp packet will have the IP address of Ethernet0 as source address
ntp source e0
ntp server IP-address-local-ISP-NTP
! restrict access to the ntp service based on the source IP address
ntp access-group 90
```

Define the ACL:

```
access-list 90 permit IP-address-local-ISP-NTP
```

In case of a NTP vulnerability, run following command in the interface e0 configuration:

```
ntp disable
```

2.2 VPN concentrator

IPsec is completely transparent to the applications. So no applications need to have any knowledge of IPsec to be able to use it. This enables GIAC Enterprises to create encrypted tunnels (VPNs).

2.2.1 Preparing OpenBSD

Edit /etc/sysctl.conf to turn ESP on and AH off on at boot time *(see 2.2.2)*

```
net.inet.esp.enable = 1
net.inet.ah.enable = 0
```

By default, OpenBSD comes with the all the necessary binaries for ISAKMP and the IPsec stack.

2.2.2 Configuring IPsec

Modes

For GIACE, IPsec provides authenticity, confidentiality and integrity of the payload network-to-network. A host-to-network to network configuration seemed to create additional maintenance efforts.

IPsec is used in tunnel mode so that the end-to-end IP header is attached to the packet, and one of the ends of the secure connection is the gateway at GIACE. This allows GIACE to tunnel the private IP address spaces over the Internet as defined in 1.6.9.

Only the ESP security protocol will be used, as the by AH authenticated IP header of the "tunneling packet" does not really add more security (the e2e IP packet is already authenticated through ESP). ESP will be configured to use SHA1 for hashing, and 3DES for encryption.

Key exchange

Although manual keying is the easiest way to get started with IPsec, GIACE prefers to go for an automated key exchange. Besides, the selected VPN devices used by the remote users do not provide any manual keying ability.
So we will use ISAKMP (Internet Security Association and Key Management Protocol). This is the standard key exchange mechanism for IPsec.

ISAKMP employs a two-phase, configurable process, for establishing the IPsec parameters between two IPsec nodes. GIACE decided that:

- The ISAKMP phase 1 communication will be in *main mode*. This will establish the Security Association while providing identity insurance. Aggressive mode might be an option in the future if more users are relying on VPN access.
  This authentication is based on a X.509 certificate generated and signed with the CA certificate by the one of the security employees using OpenSSL.
- Phase 2 will be set in *quick mode*. Quick Mode is used because there is no need to repeat a full authentication. Phase 1 has already established the SAs.

Edit /etc/isakmpd/isakmpd.conf to configure ISAKMP.

The configuration file for the GIACE VPN Concentrator and the remote office In the Netherlands are provided in Appendix.

Policy

The actual access policy is defined in /etc/isakmpd/isakmpd.policy.
This file tells ISAKMP who can access IPsec. GIACE will permit access to the VPN Access LAN for all certificates that were signed by the CA (the CA cert is pasted into this file):

```
keynote-version: 2
authorizer: "POLICY"
licensees: "x509-base64:\
      MIIBsFjFgsdfgAZERAsdgDhDFGJghkjFGKHGjkIUPyuRUITYuRTYURTyERUhTYuRhIT\
      tzERtZERTSDvxCvXCvbCVBNbvNFGjGJSDFGAZERgdfGsdFgDfGdsyErYEgDFhGFj \
      edAGjGJSDFGAZERgdfGsglSdfGSDfGsDfGVByjUUIOASjeEdfGsglSdfGSdfGsglSdf\
      FG GjGJSDFGAZERgdfGsdFgDfGdsyGjGJSDFGAZERgdfGsmdqsDqsdQSdsvSfGz\
      GHJGHQDFgqsdfQsdFSDftgAZeRTAZEGFXbvcvBFGHyujGHKGjLKyOTIYHTfNEfTf \
      tzERtZERTSDvxCvXCvbCVBNbvNFGjGJSDFGAZERgdfGsdFgDfGdsyErYEgDFhGFj \
      dFglSdfGSDfGsDfGVByjUUIOvDlmSQXCQSXlmQSCFvGùlBFGNGHNgDFfdgdFdfhd \
      tzERtZERTSDvxCvXCvbCVBNbvNFGjGJSDFGAZERgdfGsdFgDfGdsyErYEgDFhGFj \
      FSDftgAZeRTAZEGFXbvcvBFGHyujVBNbvNFGjGJSDFGAZERgdVBNbvNFGjGJdsq\
      FivHrUz7="
```

```
conditions: app_domain == "IPSEC policy" &&
        esp_present == "yes" &&
        esp_enc_alg != "null" &&
```

Note 1: The policy does not restrict access based on the ID ((ip)address). Usually this is done by adding in "*conditions*":

```
remote_id == "gw.giaceremotesite.com" -> "true";
```

GIACE can not restrict this way since remote users have no permanent addresses. Anyway, it is acceptable to rely on the certificate because the services that can be used through remote VPN are filtered by back-end firewall, and is limited to SSH/SFTP, SMTP and POP3 access.

Note 2: isakmpd.policy will not be configured to enforce destination port filtering on traffic coming out of the IPsec tunnel. GIACE prefers to have the filtering done on the back-end firewall (through the VPN Access LAN).

2.3 Front-end Firewall

2.3.1 Rulebase description

Translation of access requirements (through front-end firewall)

(see Section 1.4.5)

Rule 1	Allow the Internet community to connect to GIACE website through HTTP and HTPPS.
Rule 2	Allow the fixed IP address of “Special Partner” to connect to the web server using SFTP. Log those connections.
Rule 3	Allow any IP address to establish a IPsec connection to the VPN concentrator. (ESP and IKE)
Rule 4	Allow all remote employees (remote office/remote users/mobile users) to connect to mail server for email retrieval and sending.
Rule 5	Allow mail server to query ISP DNS,
Rule 6	MTAs on the Internet to connect to internal mail server for SMTP.
Rule 7	And the internal mail server to any server for Internet mail delivery.
Rule 8	Allow all remote employees (remote office/remote users/mobile users) to connect to the internal file server through SFTP.
Rule 9	Allow all remote employees (remote office/remote users/mobile users) to query the internal domain name server.
Rule 10	Allow all regular internal users to use HTTP and HTTPS to the Internet.
Rule 11	Allow all regular internal users to query the ISP DNS.
Rule 12	Allow all regular internal users to “ping” any IP on the Internet.
Rule 13	Allow some internal users to use passive mode FTP to the Internet. The ftp data-session must be opened dynamically by the firewall.
Rule 14	Allow security to telnet to border router.
Rule 15	Allow security to ping border router.
Rule 16	Allow syslog messages send from router to syslog server.
Rule 17	Firewall can be managed from FW-management station through SSH.
Rule 18	The NTS server must synchronize its system clock with the time provided by the router’s time server.
Rule 19	All traffic that is not explicitly allowed must be denied.

Rulebase description overview (for front-end firewall)

N°	Source	Destination	Port	Action	Log
1	Any	f.g.h.11	80/tcp, 443/tcp	Permit	N
2	j.k.l.m	f.g.h.11	22/tcp	Permit	Y
3	Any	f.g.h.10	ESP, 500/udp	Permit	N
4	192.168.10.16/28 192.168.20.0/24	192.168.1.20	25/tcp, 110/tcp	Permit	Y
5	192.168.1.20	IP_DNS_ISP	53/udp	Permit	N
6	Any	f.g.h.12	25/tcp	Permit	Y
7	192.168.1.20	Any	25/tcp	Permit	Y
8	192.168.10.16/28 192.168.20.0/24	192.168.1.51	22/tcp	Permit	N
9	192.168.10.16/28 192.168.20.0/24	192.168.1.50	53/udp	Permit	N
10	192.168.2.0/24	Any	80/tcp, 443/tcp	Permit	Y
11	192.168.2.0/24	Any	53/udp	Permit	Y
12	192.168.2.0/24	Any	ICMP echo-request	Permit	Y
13	192.168.2.12 192.168.2.20 192.168.2.21	Any	21/tcp	Permit	N
14	192.168.1.64/26	f.g.h.9	23/tcp	Permit	Y
15	192.168.1.64/26	f.g.h.9	ICMP echo-request	Permit	Y
16	f.g.h.9	f.g.h.10 -> 192.168.1.67	514/udp	Permit	Y
17	192.168.1.68	192.168.1.17	22/tcp	Permit	Y
18	192.168.1.50	f.g.h.9	123/udp	Permit	N
19	Any	Any	Any	Drop	Y

2.3.2 Preparing OpenBSD

Before the device is used a firewall, we need to

- apply latest security patches
- disable unneeded services in */etc/inetd.conf* for the "small services" and edit */etc/rc.conf to disable* portmap, sendmail and ntpd daemons
- set up routing
- enable NAT
- enable PF
- configure PF

To enable NAT-ing, edit /etc/sysctl.conf and set the line to:

```
net.inet.ip.forwarding=1
```

To enable PF, edit /etc/rc.conf and set the line to:

```
pf=YES
```

Once the system is rebooted, the configuration of PF can be started.

2.3.3 Configuring PF

We will be able to configure PF through a single file named pf.conf:

```
vi /etc/pf.conf
```

The */etc/pf.conf* file consists out of four parts:

- macro definitions / variables

- options to control how PF works
- a “scrub” line
- NAT and redirection rules
- filtering rules

None of these sections are required to exist, but those that do, must be in the above order. The syntax rules for *pf.conf* can be found in Appendix.

Options and variable definitions

Variables can be define so that when an address changes, it only need to be changed once – it also makes reading easier. When running PF, all these variables will be replaced by the value that is behind it:

```
#-------------------
# VARIABLES
#-------------------     to shorten and improve readability of rules
KS = "keep state"
# with keep state, the processed packet will be recorded into PF’s state table,
# so that valid returning packet are passed without matching a written rule.

# Interfaces – If_Ext for Internet – If_Int for internal (dmz)
If_Ext="vr0"
If_Int=”vr1”
If_All="{" $If_Ext $If_Int "}"

# Networks
Net_NotRouted="{192.168.0.0/16, 127.0.0.0/8, 172.16.0.0/12, 10.0.0.0/8 }"
Net_Internal="192.168.2.0/24"
Net_Security="192.168.1.64/26"
Net_RemoteOfficeNL=”192.168.10.16/28
Net_RemoteUsers=”192.168.20.0/24”
Net_VpnUsers=”{“$Net_RemoteUsers”, “$Net_RemoteOfficeNL”}”

# Hosts
Sv_Web="192.168.10.3"
Sv_WebPublic=” f.g.h.11”
Sv_WebPartner="192.168.10.3"
Sv_WebPartnerPublic=” f.g.h.13”
Sv_Email=”192.168.1.20”
Sv_EmailPublic=”f.g.h.12”
Sv_InternalDns=”192.168.1.50”
Sv_Files=”192.168.1.51”
Sv_Squid=”127.0.0.1”
Sv_IspDns=”t.u.v.w”
Sv_BorderRouter=”f.g.h.9”
Sv_Syslog=”192.168.1.67”
Sv_Nts=”192.168.1.50”
Pc_FwAdmin=” 192.168.1.68”
Pc_FtpUsers=”{192.168.2.12, 192.168.2.20, 192.168.2.21,}”
SpecialPartner1=”j.k.l.m”

# Services
Svc_Http="80"
Svc_Https="443"
Svc_Web="{80, 443}"
```

```
Svc_Ftp="21"
Svc_Smtp="25"
Svc_Pop3="110"
Svc_Email="{25,110}"
Svc_Telnet="23"

Svc_Dns="53"
Svc_Syslog="514"
Svc_Ssh="22"
Svc_Squid="2003"

# OPTIONS
# I did not specify any option
```

Scrub

Scrub reprocesses packets to normalize and defragment them. We will apply it on all packet entering the firewall:

```
#-----------------
# SCRUB (does not support { } )
#-----------------
scrub in on $If_Ext all
scrub in on $If_Int all
```

NAT and redirection

NAT allows the internal machines to access the Internet through one IP address. Redirection allows incoming requests to be forwarded to one of the public servers behind the NAT. This is defined by:

```
#-----------------
# NATTING
#-----------------
# GIAC Enterprises accessing Internet – Hide NAT
nat on $If_Ext from $Net_Internal to any -> $If_Ext

# Customer connections – Static NAT
        # without Squid:
rdr on $If_Ext inet proto tcp from any to $Sv_WebPublic port $Svc_Http -> $Sv_Web port
        $Svc_Http
rdr on $If_Ext inet proto tcp from any to $Sv_WebPublic port $Svc_Https -> $Sv_Web port
        $Svc_Https
        # with Squid:
#rdr on $If_Ext inet proto tcp from any to $Sv_WebPublic port $Svc_Http -> $Sv_Squid port
        $Svc_Squid
#rdr on $If_Ext inet proto tcp from any to $Sv_WebPublic port $Svc_Https -> $Sv_Squid port
        $Svc_Squid

# Customer accessing Partner Web
rdr on $If_Ext inet proto tcp from any to $Sv_WebPartnerPublic port $Svc_Http -> $Sv_Web port
        $Svc_Http
rdr on $If_Ext inet proto tcp from any to $Sv_WebPartnerPublic port $Svc_Https -> $Sv_Web
        port $Svc_Https

# Email
```

```
rdr on $If_Ext inet proto tcp from any to $Sv_EmailPublic port $Svc_Smtp -> $Sv_Email port
        $Svc_Smtp
rdr on $If_Ext inet proto tcp from any to $Sv_EmailPublic port $Svc_Smtp -> $Sv_Email port
        $Svc_Smtp
# Syslog from router
rdr on $If_Ext inet proto udp from $Sv_BorderRouter to $If_Ext port $Svc_Syslog -> $Sv_Syslog
        port $Svc_Syslog

# Employee surfing Http through Squid
#rdr on $If_Int inet proto tcp from any to any port $Svc_Http -> $Sv_Squid port $Svc_Squid
```

Filtering rules

The following lines allow the selective filtering or blocking of packets as they pass through any of the interfaces.

Note that instead of applying "last matching rule wins" filtering rules, PF will be configured to drop the packet at the first rule that matches using the "quick" option. This will enable GIAC Enterprises to build and easy readable and maintainable firewall rulebase.

For highest security, each packet passing the firewall will be filtered twice: First on the interface on which it enters, then, on the interface where it exits the firewall.

```
#--------------------
# Anti-spoofing
#--------------------
block in    log  quick on $If_Int                from ! $Net_Internal to any
block out   log  quick on $If_Int                from any to ! $Net_Internal
# Do not allow non-routable addresses passing over the Internet interface (in- and outbound)
block in    log  quick on $If_Ext                from $Net_NotRouted to any
block in    log  quick on $If_Ext                from any to $Net_NotRouted
block out   log  quick on $If_Ext                from $Net_NotRouted to any
block out   log  quick on $If_Ext                from any to $Net_NotRouted

#-----------------
# RULEBASE
#-----------------

# Squid does name resolving
pass out log    quick on $If_Ext inet proto udp from $If_Ext to $Sv_IspDns port $Svc_Dns $KS

# Rule1
        # http not through Squid:
#pass in        quick on $If_Ext inet proto tcp from ! $Net_Internal to $Sv_WebPublic port $Svc_Web
        $KS
#pass out       quick on $If_Int inet proto tcp from $If_Int to $Sv_Web port $Svc_Web $KS
        # http through squid:
pass in     quick on $If_Ext inet proto tcp      from ! $Net_Internal to $Sv_Squid port $Svc_Squid
        $KS
pass out        quick on $If_Int inet proto tcp  from $If_Int  to $Sv_Web port $Svc_Web $KS

# Rule2
pass in log     quick on $If_Ext inet proto tcp from $SpecialPartner1 to $Sv_WebPublic port
        $Svc_Ssh $KS
pass out log    quick on $If_Int inet proto tcp from $If_Int to $Sv_Web port $Svc_Ssh $KS
```

```
# Rule3
# Allow incoming VPN connections
pass in          quick on $If_Ext inet proto esp from any to $If_Ext $KS
pass in          quick on $If_Ext inet proto udp from any to $If_Ext port 500 $KS

# Rule4
pass out log     quick on $If_Int inet proto tcp from $Net_VpnUsers to $Sv_Email port $Svc_Email
          $KS

# Rule5
pass in          quick on $If_Int inet proto udp from $Sv_Email to $Sv_IspDns port $Svc_Dns $KS
pass out         quick on $If_Ext inet proto udp from $Sv_Email to $Sv_IspDns port $Svc_Dns $KS

# Rule6+7
pass in log      quick on $If_Ext inet proto tcp from any to $Sv_Files port $Svc_Ssh $KS
pass out log     quick on $If_Int inet proto tcp from $If_Int to $Sv_Files port $Svc_Ssh $KS

pass in log      quick on $If_Int inet proto tcp from $Sv_Email to any port $Svc_Smtp $KS
pass out log     quick on $If_Ext inet proto tcp from $If_Ext to any port $Svc_Smtp $KS

# Rule8+9
pass out         quick on $If_Int inet proto tcp from $Net_VpnUsers to $Sv_Email port $Svc_Email
          $KS
pass out         quick on $If_Int inet proto udp from $Net_VpnUsers to $Sv_InternalDns port
          $Svc_Dns $KS

# Rule10
# web access without Squid:
#pass in log     quick on $If_Int inet proto tcp  from $Net_Internal  to any port $Svc_Web $KS
#pass out log    quick on $If_Ext inet proto tcp from $If_Ext to any port $Svc_Web $KS
# web access with Squid:
pass in log      quick on $If_Int inet proto tcp  from $Net_Internal  to $Sv_Squid port $Svc_Squid $KS
pass out log     quick on $If_Ext inet proto tcp from $If_Ext to any port $Svc_Web $KS

# Rule11
pass in log      quick on $If_Int inet proto udp from $Net_Internal to $Sv_IspDns port $Svc_Dns $KS
pass out log     quick on $If_Ext inet proto udp from $Net_Internal to $Sv_IspDns port $Svc_Dns $KS

# Rule12
# ICMP is filtered stateful (see further: PF and ICMP). Request/replies can pass the border router.
pass in log      quick on $If_Int inet proto icmp from $Net_Internal  to any icmp-type echoreq $KS
pass out log     quick on $If_Ext inet proto icmp from $If_Ext to any icmp-type echoreq $KS

# Rule13
pass in          quick on $If_Int inet proto tcp  from $Pc_FtpUsers to any port $Svc_Ftp $KS
pass out         quick on $If_Ext inet proto tcp from $If_Ext to any port $Svc_Ftp $KS

# Rule14+15
pass in log      quick on $If_Int inet proto tcp  from $Net_Security to $Sv_BorderRouter port
          $Svc_Telnet $KS
pass out log     quick on $If_Ext inet proto tcp from $If_Ext to $Sv_BorderRouter port $Svc_Telnet
          $KS

pass in log      quick on $If_Int inet proto icmp from $Net_Security to $Sv_BorderRouter icmp-type
          echoreq $KS
```

```
pass out log    quick on $If_Ext inet proto icmp from $If_Ext to $Sv_BorderRouter icmp-type echoreq
        $KS

# Rule16
pass in log     quick on $If_Ext inet proto udp from $Sv_BorderRouter to $If_Ext port $Svc_Syslog
        $KS
pass out log    quick on $If_Int inet proto udp from $If_Int to $Sv_Syslog port $Svc_Syslog $KS

# Rule17
pass in  log    quick on $If_Int inet proto tcp  from $Pc_FwAdmin to $If_Int port $Svc_Ssh $KS

# Rule18
pass in         quick on $If_Int inet proto udp from $Sv_Nts to $Sv_BorderRouter port $Svc_Ntp $KS
pass out        quick on $If_Ext inet proto udp from $If_Ext to $Sv_BorderRouter port $Svc_Ntp $KS

# Rule19
block in    log  quick on $If_All               from any to any
block out   log  quick on $If_All               from any to any
```

Note: The man page says: "in or out - This rule applies to incoming or outgoing packets. If neither in nor out are specified, the rule will match packets in both directions."
It did not work for me however!

PF and ICMP

The way PF is able to handle ICMP traffic shows again how strong and mature this firewall is. PF is capable of dealing with both ICMP categories:

- For ICMP queries, ICMP states can be created, and PF will know how to match ICMP replies to states.
- ICMP error messages (which always refer to a TCP or UDP packet) are passed if they matched a state that was created for that TCP/UDP connection. So, if a state was created for a TCP connection, and an ICMP source quench message referring to this TCP connection arrives, it will be matched to the right state and get passed.

2.3.4 Loading the rules

To enable PF, and load natting/filtering rules defined in /etc/pf.conf:

```
pfctl -e -f /etc/pf.conf
```

To disable PF:

```
pfctl -d
```

When PF is disabled, internet access from the internal networks will not be possible as natting is disabled. The firewall host will be reachable from Internet.

2.3.5 Checking PF logs

Logged packets are sent to the pflog0 interface which is monitored by pflogd. When pflogd receives packets, it will dump the packets in /var/log/pflog in a tcpdump binary format.

So, the packet sniffing tool tcpdump can be used to read the pf log:

```
tcpdump -n -e -ttt -r /var/log/pflog
```

Because all packets are sent to an interface (pflog0), it is possible look at the logging in real-time like with a ethernet interface:

```
tcpdump -i pflog0
```

All the usual tcpdump parameters can be used to focus on some traffic streams.

2.3.6 Packet logging through syslog

With the script in the Appendices, the PF logs can easily be sent to the syslog server.

2.3.7 Using other features

To find out the “hit" statistics for each rule in /etc/pf.conf:

```
/sbin/pfctl -s rules -v
```

Show filter information (statistics and counters):

```
pfctl -s info
```

Flush current filter rules & reload:

```
/sbin/pfctl -F rules && /sbin/pfctl -R /etc/pf.conf
```

To display the current list of active MAP/Redirect filters and active sessions:

```
/sbin/pfctl -s state
```

3. VERIFY THE FIREWALL POLICY

3.1 Introduction

GIAC Enterprises knows that security can not be a one-stop-shopping thing. Their Security Policy states that all critical systems must run continuously some kind of file integrity checking so that in case of a intrusion, so that it is known what portions of the system have been compromised. Furthermore, on a regular basis technical audits have to be performed to verify that the security policies are correctly enforced.

3.2 Audit Plan

3.2.1 Goal

GIACE would like to get a confirmation that the primary firewall is indeed enforcing the correct access policies. It must also show that logging properly set up so that abnormal connection can be identified.

This will be done by the two security people immediately after the deployment of the new firewall. Once deployed, this process will be repeated every two months, or when a configuration got changed . This will always be during one of the two weekly maintenance windows (1st and 3rd Sunday of the month from 5am to 8am)

3.2.2 Considerations

The tests will be limited to non-destructive tests because those tests must be performed on the live systems. Thus no actual vulnerabilities will be exploited. It must be on the live system, because otherwise, GIACE has still no real confirmation of the live rulebase.

Usually software updates and patches are installed during those 3 hours, on this day, it is desirable that no changes are being performed on the infrastructure.

Besides the additional working hours of two employees on a Sunday morning, no additional costs are expected.

3.2.3 Scenario

To be sure that the primary firewall is configured correctly, different kind packets will be (tried to) send through the firewall, and that from all different LANs that the firewall is directly connected to. All outputs should be written to files, so that the results can be analyzed in depth after the tests.

This is what should be done:

(A) Verifying packet filtering policy and anti-spoofing

1) Bring the Internet link down

2) Connect a “packet crafter” to external network, and a “network sniffing device” to the networks on the other sides of the firewall.

3) Check if PF is correctly configured to drop traffic that should not allowed.
This by generating ICMP, UDP and TCP (with SYN bit set) packets originating from:
all GIACs public IP addresses
a random Internet routable IP address

192.168.1.x
192.168.2.x
192.168.111.6
192.168.10.x
192.168.20.x

directed to:

192.168.1.x
192.168.2.x
192.168.111.6
192.168.10.x
192.168.20.x

on following "interesting" ports for the UDP and TCP packets:

1-23,25,37, 53,79,80,87,109-111,119,389,443,512-515,540,
1080, 2000,2003,2049,6000,8000,8080,8888 on TCP
1-20,37,53, 69,111,123,161-162,514,2000,2049,6000, on UDP

4) With the packet sniffer on the other sides of the firewall, look if any of these packets are able to pass through the firewall.

5) A similar exercise should be done in the opposite way, i.e. deploying the sniffer on the external network, and send crafted packets from the Internet Services LAN to external IP addresses.
Packets with source IP:

192.168.1.x
192.168.2.x
192.168.111.6
192.168.10.x
192.168.20.x

and direct it to:

GIACs public IP addresses
random Internet addresses

and check if something passed the firewall by looking at the packet sniffer on the external network.

(B) Verify the functionality of PF

6) Connect a "packet generator" to external network, and a "network sniffing device" to the networks on the other sides of the firewall.

7) Check if PF is correctly configured to: stateful filter icmp, udp and tcp and if PF is able to process fragmented packets.

This by sending:

ICMP replies/error messages
UDP packets coming from the IP-DNS- ISP source port 53
TCP packets with ACK bit set
Fragmented TCP packets with SYN bit set

originating from:

some GIACE public IP addresses
the IP address of the ISP DNS
some random Internet routable IP address

using following source ports for the TCP packets:

21, 25, 80, 443, 2003, 57341 on TCP

using following source ports for the UDP packets:

53, 69, 123, 2000, 45556 on UDP

directed to:
f.g.h.11
f.g.h.12
a Internal LAN PC with FTP access, e.g. 192.168.2.30
on destination ports higher than 1023.

(C) Verify logging configuration – Tracking of unauthorized traffic

8) Look at the sniffed packets that were captured since the beginning of the scan. Check the policy if the capture traffic should be blocked or not. This is the most important part of the test.

3.2.4 GIACE approved supporting tools

The tools that will be used to perform the above checks are:
Nmap 3.20 – to craft packets
Hping2 2.0.0-rc2 – to craft packets
Tcpdump 3.7.2 – to sniff the network
Netcat 1.10 – to test some udp/tcp communications

They will all be compiled and executed on Debian GNU/Linux.

3.3 Conducting the Front-end Firewall Audit

Nmap and *hping2* are used to generate the traffic. *Nmap* for the TCP and UDP traffic generation, and will be used with the "-T insane" option to make it generate packets as fast as possible. We work locally on the network and we do not mind missing a response (as *tcpdump* should capture the initially sent packet). *Nmap* was selected for this purpose as it is interesting to see what information a hacker using *nmap* could see while targeting GIACE. If *nmap* is reporting the wrong information, the better for GIACE.
Hping2 will be used to craft ICMP messages.

Sending packets from outside the perimeter

Screenshots (using root account) (host located outside perimeter)	TCPDUMP screenshots (host located inside perimeter)
Launching nmap for (A) udp and tcp with SYN	Launching tcpdump (root)
What?	
Run in verbose mode, do not do reverse DNS resolution, do not ping the hosts before scanning them, send out the packets as fast as possible on interface eth0. Those packets should be UDP and a TCP (with SYN bit set) packets from source IP x.x.x.x to all hosts mentioned in the "TargetHostsFile" file on the ports specified.	Show all packets that arrive on the network.
How?	
packetgenerator:~# nmap -v –n –P0 –T insane –e eth0 –sU –sS –S SourceIP –iL TargetHostsFile –p 1-23,25,37,53,69,79,80,87,109-111,119,123,161-162, 389, 443, 512-515, 540, 1080, 2000, 2003, 2049, 6000, 8000, 8080, 8888	*# tcpdump –i eth0*
Result?	
Starting nmap V. 3.20 (www.insecure.org/nmap)	*device eth0 entered promiscuous mode*

Host (xxxxxx) appears to be up … good. *device eth0 entered promiscuous mode* *Initiating SYN Stealth Scan against (xxxxxx)* *Adding open port 25/tcp* *Adding open port 80/tcp* *Adding open port 443/tcp* *The SYN Stealth Scan took 2 seconds to scan 30 ports.* *Initiating UDP Scan against (xxxxxx)* *Adding open port 1/udp* *Adding open port 2/udp* *Adding open port 3/udp* *Adding open port xxx/udp* *[…]* *The UDP Scan took 3 seconds to scan 30 ports.* *Interesting ports on (xxxxxx):* *Port State Service* *25/tcp open smtp* *80/tcp open http* *443/tcp open https* *Nmap run completed – x IP addresses (x host up) scanned in 6 seconds*	*tcpdump: listening on eth0* *16:29:07.160000 x.x.x.x.36035 > 192.168.1.20.smtp: S 3637910958:3637910958(0) win 2048* *16:29:07.160000 192.168.1.20.smtp > x.x.x.x.40470: S 2864428697:2864428697(0) ack 3637910959 win 9112 <mss 536> (DF)* *16:29:07.160000 x.x.x.x.40470 > 192.168.1.20.smtp: R 3637910959:3637910959(0) win 0* *16:29:12.650000 f.g.h.9.43677 > 192.168.1.67.syslog: udp 0* *16:29:12.970000 f.g.h.43678 > 192.168.1.67.syslog: udp 0*
So?	
SMTP and HTTP(S) are listening. This suites the policy. Nmap though that all udp ports were open. This is due to the firewall silently dropping each udp packet without sending an icmp error message.	We do not see http(s) since it was proxied on the firewall (Good!). There was a hit on syslog (from router to syslog only); something that nmap was not able to identify. It suites the policy.
Launching hping2 for (A) ICMP	
What?	
On eth0, send an ICMP Source Quench (-K 4) from a spoofed Internet address, packet to GIACE internal user PC	Sniff the LAN if an ICMP Source Quench can pass the firewall
How?	
hping2 –I eth0 –1–K 4 –a x.x.x.x 192.168.2.5	*# tcpdump –i eth0*
Result?	
	Nothing. The firewall drops it.
This must be repeated for all other combinations specified in scenario (A).	

Launching nmap for (B) tcp with ACK bit	Launching tcpdump (root)
What?	
Run in verbose mode, do not do reverse DNS resolution, do not ping the hosts before scanning them, send out the packets as fast as possible on interface eth0. Those should be TCP packets (with ACK bit set) from source IP x.x.x.x with source port xxx to all hosts mentioned in the "TargetHostsFile" file on the ports specified.	Show all packets that arrive on the network.
How?	
packetgenerator:~# nmap -v –n –P0 –T insane –e eth0 –sA –S IS.P.DN.S –g 53 –iL TargetHostsFile –p 1-21, 25, 80, 443, 2003, 57341	*# tcpdump –i eth0*
Result?	
Starting nmap V. 3.20 (www.insecure.org/nmap)	*device eth0 entered promiscuous mode*

Host (xxxxxx) appears to be up … good. *device eth0 entered promiscuous mode* *Initiating ACK Scan against (xxxxxx)* *The ACK Scan took 1 second to scan 30 ports.* *All 27 scanned ports on (xxxxxx) are: filtered* *Nmap run completed – x IP addresses (x host up) scanned in 1 second*	*tcpdump: listening on eth0*
So?	
Nmap did not get any responses back.	The firewall did not pass anything. Good! So indeed, PF is not a simple filtering router.
Launching nmap (as root) for (B) fragmented packets	
What?	
Run in verbose mode, do not do reverse DNS resolution, do not ping the hosts before scanning them, send out the packets as fast as possible on interface eth0. Those should be TCP packets (with fragmentation and SYN bit set) from source IP x.x.x.x to all hosts mentioned in the "TargetHostsFile2" file on the ports specified.	*Show all packets that arrive on the network.*
How?	
Nmap -v -n -P0 –T insane -e eth0 -f –sS –S x.x.x.x –iL TargetHostFile2 –p 1-23,25,37,53,69,79,80,87,109-111,119,123,161-162, 389, 443, 512-515, 540, 1080, 2000, 2003, 2049, 6000, 8000, 8080, 8888	
Result?	
	device eth0 entered promiscuous mode *tcpdump: listening on eth0* *16:33:02.500000 x.x.x.x.36035 > 192.168.1.20.smtp: S 3637910958:3637910958(0) win 2048* *16:33:02.500000 192.168.1.20.smtp > x.x.x.x.40470: S 2864428697:2864428697(0) ack 3637910959 win 9112 <mss 536> (DF)* *16:29:07.160000 x.x.x.x.40470 > 192.168.1.20.smtp: R 3637910959:3637910959(0) win 0*
So?	
	Similar to the SYN scan. Good: no additional traffic can pass if it is fragmented.
This must be repeated for all other combinations specified in scenario (B).	

A script is used for all different combinations.

3.4 Audit Evaluation

Nmap sometimes gave the wrong information, mainly with UDP ports (as the firewall silently drops those packets). You can not rely on it for UDP scans. Therefore tcpdump was used on the other side of the firewall. Take the syslog connectivity for example: It was only seen by the sniffer.

With Squid as proxy on the firewall we do not see the SYN scans targeted at the web server, as Squid answering them.

Results of (a) are critical, the ones of (b) is nice to know. In (a) we did not scan all ports. Only the ports that introduce risks to GIACE. Furthermore, seen the short rulebase of the front-end firewall, there is no added value in scanning all 2x 65535 ports.

Conclusion: the firewall is configured correctly.

4. DESIGN UNDER FIRE

4.1 Selected Design

For the "Design under attack" assignment, I selected the practical of Fabio Cerniglia.

> Practical: "SANS GIAC Certified Firewall Analyst Practical Assignment"
> URL: http://www.giac.org/practical/GCFW/Fabio_Cerniglia_GCFW.pdf
> Student name: Fabio Cerniglia
> Analyst number: 0379

4.2 Firewall Attack

The firewall protecting all networks is a Cisco PIX 515E. It is running firmware version 6.2. This device can deal with 125000 simultaneous sessions. Besides functioning as a stateful filter, it will also concentrate VPN tunnels.

4.2.1 Findings concerning border router

The border router is configured with ingress filtering. With a good ingress filter, GIACE can prevent a successful attack targeted to the PIX firewall.

Looking at the border router configuration, the ingress filter only allows certain TCP sessions, to some Internet services, with the SYN bit set. This is good. One rule, however, makes the router pass all TCP traffic -with ACK bit set- from any device to any device (including the PIX or itself). This due to following line:

```
access-list 101 permit tcp any any established log
```

This allows a hacker to see which services are running on this border router, e.g. through a "nmap –sA" scan.

I also believe that the proposed router ACL is missing some permit rules to allow IPsec communications to the PIX.

4.2.2 PIX vulnerabilities

Following web sites have been searched for vulnerabilities on PIX515E:
* http://www.cisco.com/warp/public/707/advisory.html
* http://www.securityfocus.com/cgi-bin/sfonline/vulns.pl (bugtraq)
* http://neworder.box.sk
* http://www.cert.org

I found four vulnerabilities on Cisco PIX that are publicly known:
- Malformed SNMP Message-Handling Vulnerabilities (VU#617947, VU#107186)
- Scanning for SSH Can Cause a Crash (VU#945216)
- PIX does not delete duplicate ISAKMP SAs with the peer (CSCdv83490)
- Buffer overflow while doing HTTP traffic authentication using TACACS+ or RADIUS. (CSCdx35823)

Bugtraq, reported also "Cisco PIX Firewall Telnet/SSH Subnet Handling Denial Of Service Vulnerability: *"A vulnerability in the TCP/IP stack allow a remote attacker run a denial of service attack against the PIX firewall. This vulnerability is due to a wrong handling of the subnet address by the PIX OS stack. If the SSH or telnet daemon is used, the PIX will even answer to connection*

request sent to the subnet address. A DDOS attack exploiting this vulnerability -by sending packets to the subnet address- may produce memory fragmentation." As Fabio's router policy specifies that direct-broadcasts are not allowed though the border router, this vulnerability can not be exploited.

The vulnerability with the highest chance of success seems "Scanning for SSH Can Cause a Crash ". A malformed SSH packet directed at the affected device can cause a reload of the device. No authentication is necessary for the packet to be received by the affected device. See "http://www.securityfocus.com/bid/6110", "http://www.cisco.com/warp/public/707/SSH-scanning.shtml" and "http://neworder.box.sk/showme.php3?id=7519"

4.2.3 Performing the attack

Malformed packets can be generated using the SSHredder test suite from Rapid7, Inc. (Downloadable from http://www.rapid7.com, under BSD license)
It has a suite of crafted packets to test implementations of the SSH protocol. If the SSH server has been enabled, several of the test cases cause a forced reload of the device before the authentication process is called. Each time an SSH connection attempt is made with one of the crafted packets, and the SSH server is enabled on the device, the device reboots.

The "SSHredder SSH protocol test suite" contains hundreds of sample SSH packets. These invalid and/or atypical SSH packets focus on the greeting and KEXINIT (key exchange initialization) phases of SSH connections.

Once this tool is installed, the firewall (or even border router) can be attacked with those special crafted SSH packets. When successful, the PIX will keep rebooting every time SSH starts listening. The tool will be used to repeatedly generate those malformed SSH packets.

4.2.4 The defense

Workarounds consist of disabling the SSH server, removing SSH as a remote access method, permitting only trusted hosts to connect to the server, and blocking SSH traffic to the device completely via external mechanisms. The ACLs did not allow incoming SSH connections to the PIX. So, the attack is supposed to fail, but, you never know it for sure until you tried it yourself.

An upgrade might fix this vulnerability. Upgrades are available from Cisco and can be obtained through the Software Center at http://www.cisco.com/tacpage/sw-center/.

4.3 Denial of Service Attack

4.3.1 Attack scenario

The design is subjected to a DoS attack trying to saturate the WAN link, by filling up the available bandwidth between the border router and the ISP.
One slow dial-up connection will not be able to do this, but, if 50 hosts with a cable/ADSL modem start generating the maximum traffic this will not be a problem.

Since it is not trivial to have 50 cable/ADSL connections at our disposal, we will rely on badly protected PCs residing on the Internet on which we install a little program. Those compromised hosts will be our "zombie" hosts.

On each of those zombies, this little program will generate and send as much as possible different IP packets to GIACE. When each connection has a upstream speed of 128bps, 50 of

such connections will generate 6 400bits per second. Fabio's design includes a single 2 048bits per second line. Obviously, our 50 hosts can fill this ISP link with useless traffic.

Such a distributed DoS can be devastating for the company as legitimate customers and remote employees will not be able to communicate to GIACE. Furthermore, internal employees can not access the Internet.

The attack will continue until the compromised hosts stop sending traffic (the user powers off his/her machine)

4.3.2 Preparing the attack

Most popular distributed denial of service tools are Stacheldraht, Tribe Flood Network (TFN), Trin00, etc. but they all operate under Unix. WinTrin00/Troj_Trin00 and Freak88 are Windows trojans. Freak88 can only connect up to 3 infected machines and start 65000 byte ICMP floods.

In my opinion, the average Windows users are the easiest targets. Therefore we chose WinTrin00. Now, it is just a matter of getting this Trojan on 50 machines.
We just need to send the Trojan via e-mail. Most of them run a default Outlook Express installation possibly without any form of anti-virus filtering. Furthermore most of such users act careless with received emails with executables attached.
We will send a nice flash animation to some email addresses found in some "cable user" newsgroups… Using eLiTeWrap, WinTrin00 will be wrapped together with that friendly animation.

An electronic mail message is sent to an account at a free web based e-mail service to confirm that a system has been "zombified".

4.3.3 Performing the attack

The hosts on which this "friendly animation" was executed will now act as zombies. We will run a Trin00 Master host. This one will then instruct the daemon hosts to attack a victim host. Remote control of the Trin00 master is accomplished via a TCP connection to port 27665/tcp.
The daemons have the capability to send the target host a UDP flood to GIACE.

4.3.4 The defense

Stopping the attack

In case of an attack from Trin00, TFN or Stacheldraht zombies, GIACE might stop the attack by running a tool called "Zombie Zapper" from an available dial-up line. This tool is a free, open source tool that can tell a zombie system to stop flooding.
(http://razor.bindview.com/tools/ZombieZapper_form.shtml)
It does assume various defaults used by these attack tools are still in place.

To stop a Trin00 daemon on host e.v.i.l from flooding, execute

```
zz -a 1 -v e.v.i.l
```

Or to stop WinTrin00 on network e.v.i.0 from flooding where udp source port 80 is used:

```
zz -u 80 -c e.v.i.0
```

Mitigate DDOS

A DDoS attack can be partially mitigated by doing some rate-limiting:

- Request ISP to do rate-limiting on GIACE link on their side for the downstream traffic. If it concerns a Cisco router, Committed Access Rate (CAR) can be used. The rate-limiting capability introduced by CAR, can for example limit HTTP Web traffic to 60 percent of the total link bandwidth, which ensures capacity for non-Web traffic.
- Rate-limit the upstream traffic on the primary firewall (e.g. with AltQ on OpenBSD) or border router.

4.4 Internal System Compromise through Perimeter Defense

4.4.1 Selecting a target

According to Fabio's design, IIS is hardened and is up-to-date with all the security patches. Let's assume that this is the case (although we shouldn't) and select a more interesting component. According to the design the mail server runs Exchange 2000 with SP2 on Windows 2000. The need for hardening and security updates was not expressed. So, the likelihood of succeeding in a mail server attack would be higher.

4.4.2 Selecting an exploit

http://www.securityfocus.com/cgi-bin/sfonline/vulns.pl
These are the existing vulnerabilities:

- 2002-08-06: Multiple MSRPC Denial Of Service Vulnerabilities
- 2002-08-06: Post Authorization License Exhaustion Denial Of Service Vulnerability
- 2002-06-06: Malformed Mail Attribute DoS Vulnerability (CAN-2002-0368)
- 2002-03-28: Outlook Web Access with RSA SecurID Authentication Bypass Vulnerability
- 2002-03-20: SMTP Service Malformed Command Denial of Service Vulnerability
- 2002-02-07: Inappropriate Registry Permissions Vulnerability

The malformed mail attribute DoS vulnerability seems the most feasible one to exploit, as most other vulnerabilities can not be exploited through 25/tcp.

4.4.3 Performing the attack

Introduction

To support the exchange of mail with heterogeneous systems, Exchange messages use the attributes of SMTP mail messages that are specified by RFC's 821 and 822. There is a flaw in the way Exchange 2000 handles certain malformed RFC message attributes on received mail. Upon receiving a message containing such a malformation, the flaw causes "the Store service" to consume 100% of the available CPU in processing the message. This can be used for a DoS attack.
The effects of the attack would last as long as it took for the Exchange Store service to process the message. Neither restarting the service nor rebooting the server would remedy the denial of service, because the Exchange can not skip the processing of the malformed message.

Performing the attack

We will connect to the Exchange sever on port 25/tcp and make a mail message with a specially malformed attribute, so that when the message is received and processed by "the Store service", the CPU will spike to 100%.

To pass the SMTP commands to the exchange server, we will use netcat:

```
nc -v smtp.giace.com 25
```

Once connected, we enter the commands according the SMTP Command Syntax specified in http://www.ietf.org/rfc/rfc821.txt and http://www.ietf.org/rfc/rfc822.txt. Our first command will be:

```
HELO TEST.COM
```

then we introduce, one by one, all other required commands and that malformed attribute.

And once we enter:

```
QUIT
```

this message will get processed by Exchange causing unavailability of the server.

4.4.4 The defense

There is only one solution to secure against those exploits: Use recent versions and install latest security patches.

LIST OF REFERENCES

Cerniglia, Fabio. “SANS GIAC Certified Firewall Analyst Practical Assignment”. 14-Oct-2002. URL: http://www.giac.org/practical/GCFW/Fabio_Cerniglia_GCFW.pdf

“The Debian GNU/Linux Project”, URL http://www.debian.org

The Suse Linux distribution, URL: http://www.suse.com

The OpenBSD Project, “Documentation and Frequently Asked Questions”, URL http://www.openbsd.org/faq/index.html

The OpenBSD Project, “OpenBSD Manpages”, URL http://www.openbsd.org/cgi-bin/man.cgi

Dell, “Power Edge Features”, URL: http://www.dell.com/us/en/esg/topics/segtopic_servers_pedge_rackmain.htm

A graphical interface for PF, FW-builder, URL www.fwbuilder.org

“FreeS/WAN”, URL: http://www.xs4all.nl/~freeswan/

Cisco Systems, “1700 series routers”, URL: http://www.cisco.com/warp/public/cc/pd/rt/1700/

The PartImage Project. “PartImage Disk imaging tool”, URL http://www.partimage.org

“An OpenSource OfficeSuite”. URL: http://www.openoffice.org

“Opensource Intrusion Detection”. URL: http://www.snort.org/

“OpenBSD CryptoCards”. URL: http://www.openbsd.org/crypto.html#hardware

“PowerCrypt Encryption Accelerator”. URL: http://www.powercrypt.com/#specs

The Apache Project. “Open Source web server”. URL: http://www.apache.org/

Squid. “OpenSource web proxy cache”. URL: http://www.squid-cache.org

SANS. “SANS module 3.2 and 3.1 – Securing routers”.

Keeney, Franky. ”Screening Router Access List”, URL: http://pasadena.net/cisco/secure.html

APPENDICES

A. OpenBSD Crypto Accelerators

From OpenBSD Crypto FAQ: *“Cards using the Hifn 7751 can be used as a symmetric cryptographic accelerator. Current performance using a single Hifn 7751 on each end of a tunnel is 64Mbit/sec for 3DES/SHA1 ESP, nearly a 600% improvement over using a P3/550 CPU. Further improvements are under way to resolve a few more issues, but as of April 13, 2000 the code is considered stable. We wrote our own driver for supporting this chip, rather than using the (USA-written) PowerCrypt driver, as well our driver links in properly to the IPsec stack. The 7751 is now considered slow by industry standards and many vendors have faster chips (even Hifn now has a faster but more expensive chip). Peak performance with 3DES SHA1 ESP is around 64Mbit/sec.”*

B. PowerCrypt

Card Type: 32-bit, 33 MHz half-length PCI card, 5-volt I/O specification (compatible with most PCs and motherboards)
Processor: Hifn 7751, 66 MHz
Memory: 512K static RAM; single or multiple compression contexts, encryption contexts as memory allows. Encryption contexts can be 128 or 512 bytes
Encryption Algorithms: DES, 3DES (ECB, CBC, CFB, and OFB modes for both), RC4
Authentication Algorithms: MD5 and SHA-1 in HMAC, SSL-MAC and plain digest modes
Compression Algorithms: LZS and MPPC
Throughput (3DES): 10 MB/sec minimum
Physical Measurements: New low-profile version for rack-mount and NLX applications, meauring 2.5" H x 4.69" L (63.5mm H x 118mm L)
Power Consumption: DC 5 Volts, 160mA nominal
Approvals: UL 94V-0 (PCB)
Price: $500

C. Cisco 1760

CISCO1760

- 10/100 Modular Router w/ 2WIC/VIC,2VIC slots,19 inch Chassis
- MEM1700-32U96D
  Cisco 1700 32MB to 96MB DRAM Factory Upgrade
- WIC-1T
  1-Port Serial WAN Interface Card
- WIC-1B-S/T
  1-Port ISDN WAN Interface Card(dial and leased line)
- Included: S17C-12211T
  Cisco 1700 IOS IP
- Included: CAB-AC
  Power Cord,110V

D. Dell PowerEdge 2650 Features

Processors
- Up to 2 Intel® Xeon Processors at 1.8GHz, 2GHz, 2.2GHz, 2.4GHz, 2.6GHz and 2.8GHz with NetBurst Micro-architecture with Hyper-Threading technology

Front Side Bus
- 400MHz front side bus that allows for faster data throughput than 133MHz front side bus speeds

Cache
512KB L2 Advanced Transfer Cache

Chipset
- ServerWorks GC-LE chipset supports 5 PCI buses: 3 PCI-X (1 X 64bit/133MHz, 2 X 64bit/100MHz), 1 x 64bit/66MHz, 1 legacy bus (32bit/33MHz)

Memory
- 256MB - 6GB 200MHz DDR SDRAM
- 6 DIMM sockets on system board configurable for Spare Bank

Expansion Slots
3 full length PCI-X slots (1 X 64bit/133MHz, 2 X 64bit/100MHz)

Drive Controller
- Dual-channel integrated Ultra3 (Ultra160) SCSI Adaptec® AIC-7899 (160Mb/s) controller provides latest high performance SCSI technologies available without taking up PCI slots
- Optional Adaptec Ultra3 (Ultra160) SCSI 39160 card

RAID Controller
- Optional, embedded PowerEdge Expandable RAID controller, Version 3, Dual-Channel Integrated (PERC 3/Di) with battery-backed cache (internal channels only). Activated with optional RAID Enablement Key.
- Optional, PowerEdge Expandable RAID controller, Version 3, Dual-Channel (PERC 3/DC) or Quad-Channel (PERC3/QC) with 128MB battery-backed cache

Drive Bays
- Hard Drive bays for 5 X 1" hot-plug SCSI drives
- Media bay for one 24X EIDE CD-ROM or 8X IDE DVD ROM, one 3.5" 1.44MB diskette drive
- Backplane may be split for a 2 + 3 configuration

Hard Drives
- 18GB,1 36GB, 73GB, 146GB (10,000 rpm) and 18GB, 36GB (15,000 rpm) Ultra3 (Ultra160) SCSI
- 18GB, 36GB, 146GB fibre channel (10,000 rpm)(external only)
- Maximum Internal Storage 730GB (5 X 146GB)

External Storage Options
- Optional PowerVaultTM 2xxS SCSI external storage system
- Optional PowerVault 660F, 224F, 650F, and 630F fibre channel RAID system
- Optional Fibre Channel Host Bus Adapter (Optical or Copper)
- Dell / EMC Storage Solutions

Cluster Support
- 2-node SCSI
- 2-node Fibre Channel

Tape Backup Options
- PowerVault 120T DLT1 Autoloader
- PowerVault 136T LTO, SDLT
- PowerVault 128T LTO, SDLT
- PowerVault 122T VS80
- PowerVault 112T Tape Rack Enclosure ideal for rack dense servers

Communications

- Dual Embedded BroadcomTM NetXtremeTM Gigabit Server Adapter to preserve valuable PCI slots. Embedded NICs support PXE and teaming functions like fail over and load balancing.
- Intel Pro/100+ Dual Port Server Adapter
- Intel Pro/1000XT (copper) and Intel Pro/1000F (fibre) Server Adapter
- Broadcom NetXtreme Gigabit Ethernet Server Adapter

Input Devices
- Windows keyboard

Ports
- 2 9-pin serial, 2 Universal Serial Bus, video, PS/2 mouse, PS/2 keyboard, 3 RJ45

Power
- Optional, hot plug, redundant 500 watts power supplies
- Voltage: 100-240 VAC

Availability
- Spare Bank configurable ECC memory
- Chipkill supported with 512KB and 1GB ECC memory DIMMs
- Dual channel embedded Ultra 3 RAID with battery-backed cache
- Dual embedded NICs with failover and load balancing support
- Hot-pluggable redundant power supplies and hot-plug fans
- Hot-pluggable hard drives
- High availability Fibre and SCSI cluster support
- Front mounted keyboard, video and monitor ports

Chassis
- Rack-Mountable Chassis: 3.375" (8.5725cm) H x 19.00" (48.26cm) W x 27.50" (69.85cm) D
- 2U rack height
- Active ID includes an illuminated indicator that provides basic system status information
- Front mounted keyboard, video and monitor ports provide easy access for crash cart
- Front mounted LCD alphanumeric display shows error messages and codes and illuminates different colors to indicate system status
- Cable-less motherboard design routes all internal connections through the printed wire assemblies to improve ease of serviceability (one cable in the system to connect backplane to control panel)
- Weight: Up to 55 lb.

Graphics
- Integrated ATI-Rage XL controller w/8MB of SDRAM (not upgradable)

Management
- Embedded Remote Access (ERA) allowing remote management of servers
- Pre Executable Environment (PXE) support of embedded NICs
- Fault monitoring of voltage, fan, and thermal conditions to help ensure notification in case of potential problems
- Management of drive array under optional PowerEdge Expandable RAID Controller
- Tracks memory errors that have been corrected by the ECC memory
- Automatic Server Recovery will reboot and restart the server if the OS hangs without user intervention
- User-definable OS thresholds can be set, allowing administrators to tune systems and eliminate bottlenecks to performance
- Email or paging through Dell OpenManageTM keeps administrators informed of potential server problems before they become critical
- Asset management features enable customers to inventory server configuration, CPU, memory and disk information, helping keep track of systems and keep them up-to-date

- Dell OpenManage Server Setup CD included with every server to get your PowerEdge up, running, and contributing to your infrastructure quickly

E. Isakmpd.conf for the VPN Concentrator
From the OpenBSD FAQ

```
[General]
Policy-File=            /etc/isakmpd/isakmpd.policy
Retransmits=            5
Exchange-max-time=      120
Listen-on=              f.g.h.10

# The name work-gw here is used just as a section name and a tag for
# use in this configuration file below and need not actually be the
# real hostname or domain name of the peer (but it could be). The IP
# address however needs to be correct. Phase 1, as you might already
# know, is to negotiate an ISAKMP security association (SA). There
# should of course be one IP and name for each peer we want to
# communicate with.
[Phase 1]
m.n.o.p=            work-gw

# Now phase 2 is negotiating IPSEC SAs. As in phase 1, the name here
# is a section name to be used later. Actually, it can be a comma
# separated list of section names here. Thus if traffic from many
# networks (or individual hosts) should be forwarded through this
# tunnel, more section names would be added (and of course corresponding
# new sections further down).
[Phase 2]
Connections=        work-gw-my-gw

# Now, here are some parameters for the ISAKMP SA negotiations. Almost
# self documenting. The section name is from [Phase 1] above. The most
# interesting tag might be the ID tag. The ID tag is set to the name
# of the section where the identity information about this host that
# will be presented to connecting peers, can be found. If the ID tag
# is not available, isakmpd will assume that it will identify itself
# using the IP address. You might also notice that there is no longer
# any authentication tag here in this configuration. The authentication
# data is currently used only in the pre-shared key case.
[work-gw]
Phase=              1
Transport=          udp
Local-address=      f.g.h.10        # Local address
Address=            m.n.o.p # Peer address
ID=                 my-ID
Configuration=      Default-main-mode

# This is the identity data. ID-type may also be IPV4_ADDR (the
# default), IPV4_ADDR_SUBNET or UFQDN. The Name tag is used for
# FQDN and UFQDN, for IPV4_ADDR an Address tag would be used instead.
# For IPV4_ADDR_SUBNET a Network and a Netmask tag would be used.
[my-ID]
ID-type=            FQDN
```

```
Name=               gw.giacenterprises.com

# This is the section for the IPSEC connection. The section name is
# from the list in the [Phase 2] section above. The ISAKMP-peer is,
# of course, the tag of our peer from section [Phase 1] above. The
# Local-ID and Remote-ID tags should be section names describing which
# packages should be forwarded over the IPSEC tunnel to the remote
# network.
[work-gw-my-gw]
Phase=              2
ISAKMP-peer=        work-gw
Configuration=      Default-quick-mode
Local-ID=           Net-west
Remote-ID=          Net-east

# Any packet originating from a computer on the network described
# here...
[Net-west]
ID-type=            IPV4_ADDR_SUBNET
Network=            192.168.3.0
Netmask=            255.255.255.0

# ... and with a destination matching the network described here,
# will be encrypted and forwarded over the IPSEC tunnel to the remote
# system.
[Net-east]
ID-type=            IPV4_ADDR_SUBNET
Network=            192.168.10.16
Netmask=            255.255.255.240

# Main mode descriptions

# Here are the data for main mode. Using DES here for real purposes
# is not very smart since DES is no longer considered a secure
# encryption algorithm. 3DES is generally considered to have much better
# security since it has enough bits in the key to be considered secure.
# Transforms is a list of tags describing main mode transforms. In
# this example we have only one.
[Default-main-mode]
DOI=                        IPSEC
EXCHANGE_TYPE=              ID_PROT
Transforms=                 3DES-MD5


# Certificates stored in PEM format
# This is important when using certificates. The CA certificates should
# be in the CA-directory (but not the CA private key of course).
# The Cert-directory should have at least the certificate for the
# local host but other certificates are also allowed. The private key
# should be the private key of the local host.
[X509-certificates]
CA-directory=       /etc/isakmpd/ca/
Cert-directory=     /etc/isakmpd/certs/
Private-key=        /etc/isakmpd/private/local.key

# Main mode transforms
```

```
#######################

# Here is our main mode transform. The important thing here is to use
# RSA_SIG as authentication method when using certificates. It is the
# only method supported when using certificates so far. Commercial
# entities in the US will thus have to wait until September 2000 to
# use this due to the RSA patent. Luckily, I am not living in the US.
# Also important is the GROUP_DESCRIPTION tag. It must match the
# GROUP_DESCRIPTION tag in the Quick mode transforms further down.
# The Life tag here could possibly be modified. The LIFE_60_SECS might
# be shorter than necessary for normal use.

[3DES-MD5]
ENCRYPTION_ALGORITHM=   3DES_CBC
HASH_ALGORITHM=         MD5
AUTHENTICATION_METHOD=  RSA_SIG
GROUP_DESCRIPTION=      MODP_1024
Life=                   LIFE_60_SECS,LIFE_1000_KB

# Quick mode description
#########################

[Default-quick-mode]
DOI=                    IPSEC
EXCHANGE_TYPE=          QUICK_MODE
Suites=                 QM-ESP-3DES-MD5-PFS-SUITE

# Quick mode protection suites
##############################
# 3DES

[QM-ESP-3DES-MD5-PFS-SUITE]
Protocols=              QM-ESP-3DES-MD5-PFS

# 3DES

[QM-ESP-3DES-MD5-PFS]
PROTOCOL_ID=            IPSEC_ESP
Transforms=             QM-ESP-3DES-MD5-PFS-XF

# Quick mode transforms

# Don't forget. The GROUP_DESCRIPTION must match the GROUP_DESCRIPTION
# in main mode above. For forwarding packets between two networks (or
# from a host to a network) we use TUNNEL mode. Between two hosts we
# may also use TRANSPORT mode instead.
[QM-ESP-3DES-MD5-PFS-XF]
TRANSFORM_ID=           3DES
ENCAPSULATION_MODE=     TUNNEL
AUTHENTICATION_ALGORITHM=       HMAC_MD5
GROUP_DESCRIPTION=      MODP_1024
Life=                   LIFE_60_SECS

# As we know from the isakmpd.config manpage the LIFE_DURATION here is
# an offer value (60), a minimum acceptable value (45) and a maximum
# acceptable value. The isakmpd.conf example has this set to
```

```
# 600,450/720 instead. That might be a better value for normal use.
[LIFE_60_SECS]
LIFE_TYPE=          SECONDS
LIFE_DURATION=      60,45:72

[LIFE_1000_KB]
LIFE_TYPE=          KILOBYTES
LIFE_DURATION=      1000,768:1536
```

F. PF Packet Logging through Syslog

Extract taken from the OpenBSD FAQ (http://www.openbsd.org/faq/faq6.html):

In many situations it is desirable to have the firewall logs available in ASCII format and/or to send them to a remote logging server. All this can be accomplished with 2 small shell scripts and with minor changes of the OpenBSD configuration files.

Syslogd is the standard daemon for logging, it logs in ASCII and is also able to log to a remote logging server. First we have to create a user pflogger with a .nologin. shell. The easiest way to create this user is with adduser.

After creating the user pflogger create the following two scripts:

```
/etc/pflogrotate
FILE=/home/pflogger/pflog5min.$(date "+%Y%m%d%H%M")
kill -ALRM $(cat /var/run/pflogd.pid)
if [ $(ls -l /var/log/pflog | cut -d " " -f 8) -gt 24 ]; then
        mv /var/log/pflog $FILE
        chown pflogger $FILE
        kill -HUP $(cat /var/run/pflogd.pid)
fi
```

```
/home/pflogger/pfl2sysl
#!/bin/sh
# feed rotated pflog file(s) to syslog
for logfile in /home/pflogger/pflog5min* ; do
        tcpdump -n -e -ttt -r $logfile | logger -t pf -p local0.info
        rm $logfile
done
```

Edit the cron job for user root
```
# crontab -u root -e
```

and add the following two lines:
```
# rotate pf log file every 5 minutes
0-59/5 *        *       *       *       /bin/sh /etc/pflogrotate
```

Create a cron job for user pflogger
```
# crontab -u pflogger -e
```

and add the following two lines:
```
# feed rotated pflog file(s) to syslog
0-59/5 *        *       *       *       /bin/sh /home/pflogger/pfl2sysl
```

Add the following line to /etc/syslog.conf:
local0.info /var/log/pflog.txt

If you want to log to a remote log server also add the line:
local0.info @syslogger

and make sure host syslogger has been defined in the /etc/hosts file.

All logged packets are sent to /var/log/pflog.txt. If the second line is added too they are sent to the remote logging host syslogger as well.

/etc/pflogrotate now processes and then deletes /var/log/pflog so rotation of pflog by newsyslogd(8) is no longer necessary and it should be disabled. However /var/log/pflog.txt replaces /var/log/pflog and rotation of it should be activated. Change /etc/newsyslog.conf as follows:

```
#/var/log/pflog      600   3     250   *     ZB    /var/run/pflogd.pid
/var/log/pflog.txt   600   7     *     24
```

Pf will now log in ASCII to /var/log/pflog.txt. If so configured in /etc/syslog.conf it will also log to a remote server. The logging is not immediate but it can take up to about 5-6 minutes (the cron job interval) before the logged packets appear in the file.

G. Syntax for PF filtering rules

Extract taken from the OpenBSD *pf.conf* manpage:

> The rule parameters specify the packets to which a rule applies. A packet always comes in on, or goes out through, one interface. Most parameters are optional. If a parameter is specified, the rule only applies to packets with matching attributes. Certain parameters can be expressed as lists, in which case pfctl(8) generates all needed rule combinations.
>
> in or out
> This rule applies to incoming or outgoing packets. If neither in nor out are specified, the rule will match packets in both directions.
>
> log In addition to the action specified, a log message is generated. All packets for that connection are logged, unless the keep state or modulate state options are specified, in which case only the packet that establishes the state is logged. (See keep state and modulate state below). The logged packets are sent to the pflog(4) interface. This interface is monitored by the pflogd(8) logging daemon, which dumps the logged packets to the file /var/log/pflog in pcap(3) binary format.
>
> log-all
> Used with keep state or modulate state rules to force logging of all packets for a connection. As with log, packets are logged to pflog(4).
>
> quick
> If a packet matches a rule which has the quick option set, this rule is considered the last matching rule, and evaluation of subse-

quent rules is skipped.

on _interface_
This rule applies only to packets coming in on, or going out through, this particular interface.

_af_ This rule applies only to packets of this address family. Supported values are inet and inet6.

proto _protocol_
This rule applies only to packets of this protocol. Common protocols are icmp(4), icmp6(4), tcp(4), and udp(4). For a list of all the protocol name to number mappings used by pfctl(8), see the file /etc/protocols.

from _source_ port _source_ to _dest_ port _dest_
This rule applies only to packets with the specified source and destination addresses and ports.

Addresses can be specified in CIDR notation (matching netblocks), as symbolic host names or interface names, or as any of the following keywords:

any Any address.
no -route Any address which is not currently routable.
table Any address that matches the given table.

Interface names can have modifiers appended:

:network Translates to the network(s) attached to the interface.
:broadcast Translates to the interface's broadcast address(es).

Host name resolution and interface to address translation are done at ruleset load-time. When the address of an interface (or host name) changes (under DHCP or PPP, for instance), the ruleset must be reloaded for the change to be reflected in the kernel. Surrounding the interface name in parentheses changes this behaviour. When the interface name is surrounded by parentheses, the rule is automatically updated whenever the interface changes its address. The ruleset does not need to be reloaded. This is especially useful with nat.

Ports can be specified either by number or by name. For example, port 80 can be specified as www. For a list of all port name to number mappings used by pfctl(8), see the file /etc/services.

Ports and ranges of ports are specified by using these operators:

= (equal)
!= (unequal)
< (less than)
<= (less than or equal)
> (greater than)
>= (greater than or equal)
>< (range)

<> (except range)

>< and <> are binary operators (they take two arguments), and the range does not include the limits. For instance:

port 2000 __ 2004
means `all ports > 2000 and < 2004', hence ports 2001, 2002 and 2003.

port 2000 __ 2004
means `all ports < 2000 or > 2004', hence ports 1-1999 and 2005-65535.

The host and port specifications are optional.

group _group_
Similar to user, this rule only applies to packets of sockets owned

by the specified group.

user _user_
This rule only applies to packets of sockets owned by the specified user. For outgoing connections initiated from the firewall, this is the user that opened the connection. For incoming connections to the firewall itself, this is the user that listens on the destination port. For forwarded connections, where the firewall is not a connection endpoint, the user and group are unknown.

All packets, both outgoing and incoming, of one connection are associated with the same user and group. Only TCP and UDP packets can be associated with users; for other protocols these parameters are ignored.

User and group refer to the effective (as opposed to the real) IDs, in case the socket is created by a setuid/setgid process. User and group IDs are stored when a socket is created; when a process creates a listening socket as root (for instance, by binding to a privileged port) and subsequently changes to another user ID (to drop privileges), the credentials will remain root.

User and group IDs can be specified as either numbers or names. The syntax is similar to the one for ports. The value unknown matches packets of forwarded connections. unknown can only be used with the operators = and !=. Other constructs like user >= unknown are invalid. Forwarded packets with unknown user and group ID match only rules that explicitly compare against unknown with the operators = or !=. For instance user >= 0 does not match forwarded packets.

flags _a_/_b_ | /_b_
This rule only applies to TCP packets that have the flags _a_ set out of set _b_. Flags not specified in _b_ are ignored. The flags are: (F)IN, (S)YN, (R)ST, (P)USH, (A)CK, (U)RG, (E)CE, and C(W)R.

flags S/S Flag SYN is set. The other flags are ignored.

flags S/SA Out of SYN and ACK, exactly SYN may be set. SYN, SYN+PSH and SYN+RST match, but SYN+ACK, ACK and ACK+RST do not. This is more restrictive than the previous example.

flags /SFRA
If the first set is not specified, it defaults to none. All of SYN, FIN, RST and ACK must be unset.

icmp-type _type_ code _code_

icmp6-type _type_ code _code_
This rule only applies to ICMP or ICMPv6 packets with the specified type and code. This parameter is only valid for rules that cover protocols ICMP or ICMP6. The protocol and the ICMP type indicator (icmp-type or icmp6-type) must match.

allow-opts
By default, packets which contain IP options are blocked. When allow-opts is specified for a pass rule, packets that pass the filter based on that rule (last matching) do so even if they contain IP options. For packets that match state, the rule that initially created the state is used. The implicit pass rule that is used when a packet does not match any rules does not allow IP options.

label _string_
Adds a label (name) to the rule, which can be used to identify the rule. For instance, pfctl -s labels shows per-rule statistics for rules that have labels.

The following macros can be used in labels:

$if The interface.
$srcaddr The source IP address.
$dstaddr The destination IP address.
$srcport The source port specification.
$dstport The destination port specification.
$proto The protocol name.
$nr The rule number.

The macro expansion for the label directive occurs only at configuration file parse time, not during runtime.

queue _queue_ | (_queue_, _queue_)
Packets matching this rule will be assigned to the specified queue. If two queues are given, packets which have a tos of lowdelay and TCP ACKs with no data payload will be assigned to the second one. See QUEUE RULES for setup details.