Global Information Assurance Certification Paper

Firewalls, Perimeter Protection, and VPNs
GCFW Practical Assignment
Version 1.5e

James Filiberto

Assignment 1 - Security Architecture (25 Points)

Define a security architecture for GIAC Enterprises, a growing Internet startup that expects to earn $200 million per year in online sales of fortune cookie sayings, and which has just completed a merger/acquisition. Your architecture must specify filtering routers, firewalls, VPNs to partners, secure remote access, and internal firewalls. Be explicit about the brand and version of each perimeter defense component. Produce a diagram or set of diagrams with explanatory text that define how to use perimeter technologies to implement your security architecture.

You must consider and define access for:

- Customers (the companies that purchase bulk online fortunes);
- Suppliers (the authors of fortune cookie sayings that connect to supply fortunes);
- Partners (the international partners that translate and resell fortunes).

As with any Ecommerce site, redundancy is a key consideration. But as this is a startup and financial considerations can dictate procedures, we have decided to create a fault tolerant system via backups. And as time and profits allow and traffic demands, we will eventually employ a high availability solution. The security architecture however does take into consideration the availability of additional support resources, growth and migration paths.

Using a 'Defense in Depth' approach that starts with a general access area and builds on layers of protection and detection, we employ a Border Router, which connects to a hub which has an Intrusion Detection System connected to it as well as redundant commercial firewalls. The firewalls include Virus protection and VPN modules for additional protection and access into the network. Also in between each segment is a switch that will help prevent packet sniffing

In the first level of our Defense in Depth approach, we created a screened network that consists of a public WWW server for general access, a private WWW server for partners and approved customers, the external DNS servers which only contain the addresses of the public Web and e-mail servers, an Intrusion Detection System, an SMTP server, a tape library, and a logging server. Access to this segment is through a firewall with 3 interfaces which can we can apply specific rules that apply to only this segment and a switch that will help prevent packet sniffing.

In the second level of our Defense in Depth approach, we have the general access corporate servers, an Intrusion Detection System, a time server to synchronize all servers, the internal DNS which resolves internal addresses for the private network space, and a proxy server that the private network space uses to access the Internet. Access to this segment is through a firewall with 3 interfaces which can we can apply specific rules that

apply to only this segment and a switch that will help prevent packet sniffing.

In the third level of our Defense in Depth approach, we have the secure access corporate servers and an Intrusion Detection System . Access to this segment is through a firewall with 3 interfaces which can we can apply specific rules that apply to only this segment and a switch that will help prevent packet sniffing.

And finally in the fourth level of our Defense in Depth approach, we have the super secure access database servers and an Intrusion Detection System. Access to this segment is through a firewall with 2 interfaces which can we can apply specific rules that apply to only this segment and a switch that will help prevent packet sniffing.

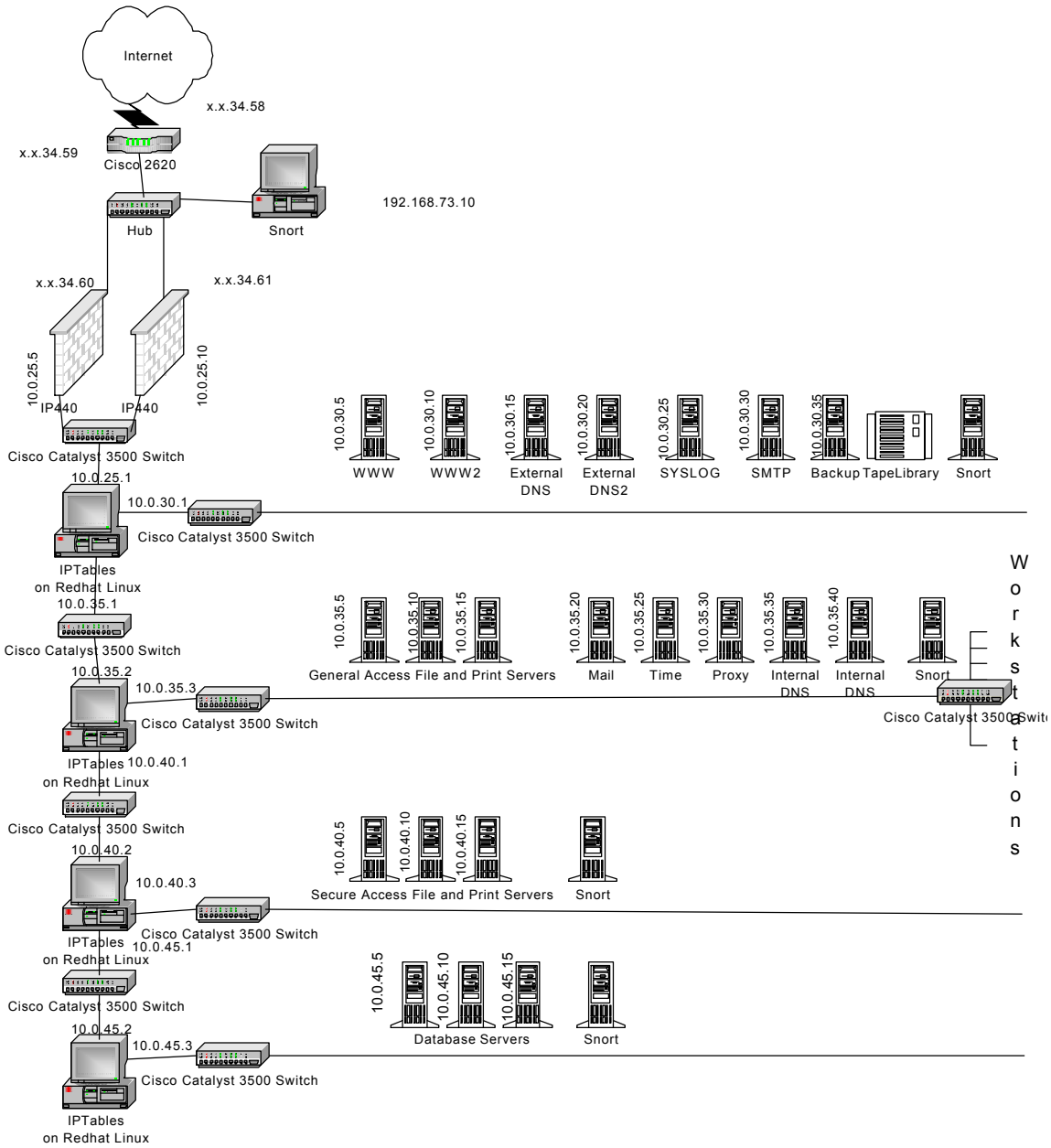
A check for new vulnerabilities and fixes is made daily, and patches applied.

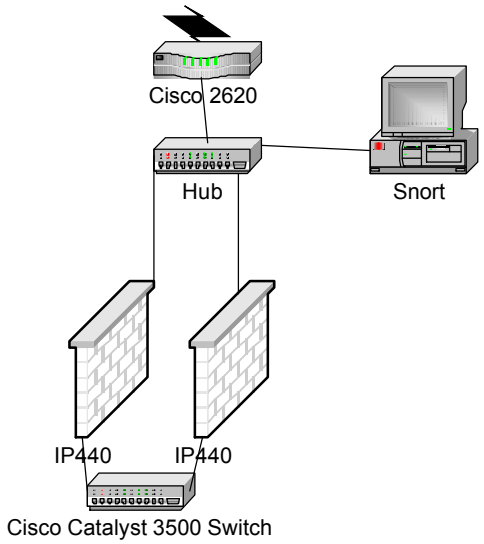
For Microsoft products:

http://www.microsoft.com/technet/treeview/default.asp?url=/technet/itsolutions/security/current.asp

For RedHat Products:

http://www.linuxsecurity.com/





In the general access area we employ a Cisco 2600 series as the Border Router. Behind the Border Router is a hub to which we have attached a Redhat Linux 7.1 box running Tripwire 2.2.1 and Snort 1.8p1. Also connected to the hub are 2 Nokia Firewall/VPN Appliances (IP440s) running Checkpoint 4.1 and WebShield for Nokia Appliance. The Nokia Firewall/VPN Appliances are connected to a Cisco Catalyst 3500 Series XL Switch.

The Border Router is the first line of defense and will be used to defend against common attacks such as IP spoofing, Smurf attacks and Source route manipulation.

Behind the Border Router is a hub which has an Intrusion Detection System connected to it as well. The Intrusion Detection System performs real-time traffic analysis and packet logging using a rules based approach.

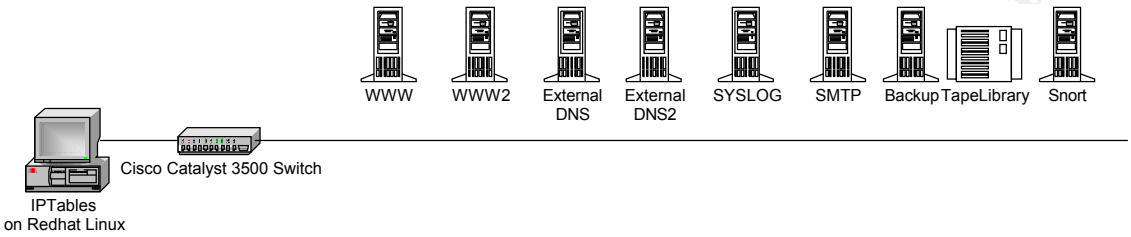
Also Connected to the hub are 2 Firewall/VPN appliances that employ real-time traffic analysis and packet logging using a rules based approach to allow or deny packets and they also employ virus scanning. There are 2 because besides giving us a migration path to redundant access of the Ecommerce sites in the future, they currently enhance the Internet connections for the VPNs that are used by our partners and remote employees.

The Firewall/VPN Appliances are connected to a Cisco Catalyst 3500 Series XL Switch, thereby greatly reducing the opportunity for packet sniffing.

The Border Router is a Cisco 2620 running IOS 12.X.
The Intrusion Detection System is a hardened Red Hat Linux 7.1 system with Tripwire 2.2.1 and Snort 1.8p1 installed. Logging is local.
The Firewall/VPN appliances are Nokia IP440s running Checkpoint Firewall 1 4.1 and WebShield for Nokia Appliance.
And a Cisco Catalyst 3500 Series XL Switch

WebShield for Nokia Appliance

http://www.nokia.com/securitysolutions/network/webshield.html



The first level, or public level consists of a Redhat Linux 7.1 box running Tripwire 2.2.1 and IPTables and has 3 interfaces. Interface 1 is connected to the Cisco Catalyst 3500 Series XL Switch which is incoming from the Internet. Interface 2 is connected to the Cisco Catalyst 3500 Series XL Switch which is connected to the segment that contains the WWW servers, External DNS servers, SMTP server, Logging Server, Intrusion Detection System, and Tape Library. Interface 3 is connected to another Cisco Catalyst 3500 Series XL Switch which proceeds further into the network.

This segment starts off with a stateful firewall (IPTABLES) running on a hardened OS that employs a program that verifies file integrity (Tripwire) and sends alerts if that verification fails. The firewall is rules based and can perform masquerading if needed.

Next is a Cisco Catalyst 3500 Series XL Switch to prevent packet sniffing.

The following servers reside on this segment:

- The first WWW server which is a general use web server that primarily delivers corporate data
- The second WWW server which is a front end for connectivity to the database
- The 2 external DNS servers
- The Intrusion Detection System for this segment
- The SMTP server
- The logging server

- The Tape Library

These servers defined:

The first WWW server which is a general use web server that primarily delivers general use corporate data is running Windows 2000 SP2 running Internet Information Server 5 and has no unneeded services running.

The second WWW server is a front end for connectivity to the database servers and is running Windows 2000 SP2 running Cold Fusion 5 and houses the scripts needed to run the SQL to access the database servers and has no unneeded services running.

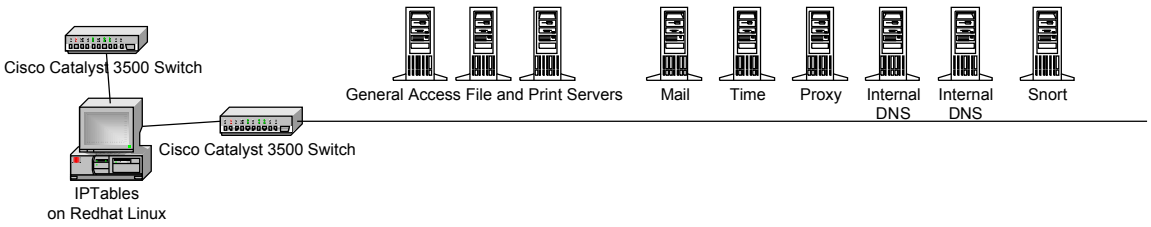
The 2 external DNS servers are running Windows 2000 SP2, and Microsofts DNS service. All other nonessential services have been turned off as have all other non essential ports.

The Intrusion Detection System is a hardened RedHat Linux 7.1 box running Tripwire 2.2.1 and Snort 1.8p1. Snort is setup to log all attempts to the logging server

The SMTP server is running on Windows NT4 SP6a running Lotus Notes v 5.05.

The logging server is running Syslogd on a RedHat Linux 7.1 box running Tripwire 2.2.1.

The Backup solution consists of a Widows 2000 Server SP2 running Legato Networker 6.0 connected via SCSI to a StorageTek Magnabox 9730 Tape Library



The second level, or General Access level consists of a Redhat Linux 7.1 box running Tripwire 2.2.1 and IPTables and has 3 interfaces. Interface 1 is connected to the Cisco Catalyst 3500 Series XL Switch which is incoming from the Internet and the First level. Interface 2 is connected to the Cisco Catalyst 3500 Series XL Switch which is connected to the segment that contains the General Access file and print servers, a time server, the internal DNS, a proxy server, a mail server and an Intrusion Detection System. Interface 3 is connected to another Cisco Catalyst 3500 Series XL Switch which proceeds further into the network.

This segment starts off with a stateful firewall (IPTABLES) running on a hardened OS that employs a program that verifies file integrity (Tripwire) and sends alerts if that verification fails. The firewall is rules based and can perform masquerading if needed.

Next is a Cisco Catalyst 3500 Series XL Switch to prevent packet sniffing.

The following servers reside on this segment:

- File Servers
- Print Servers
- Time Server
- Internal DNS Server
- Proxy Server
- Mail Server
- Intrusion Detection System

These servers defined:

The File and Print Servers are running Windows 2000 SP2 using Active Directory.

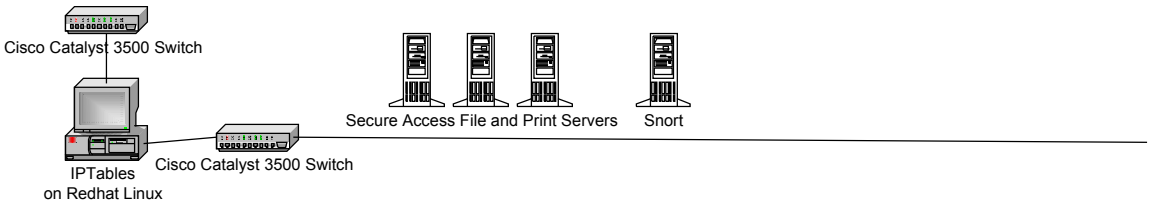
The Times Server is running Windows 2000 SP2 running Tardis 2000 V1.3.

The DNS Server is running Windows 2000 SP2 with Microsoft DNS Service

The Proxy Server is running Windows NT 4 SP6a with Microsoft Proxy Server 2.0 and WebShield Proxy for NT (http://download.nai.com/products/media/mcafeeb2b/support/wsp40nug.pdf)

The Mail server is running on Windows NT4 SP6a running Lotus Notes v 5.05.

The Intrusion Detection System is a hardened RedHat Linux 7.1 box running Tripwire 2.2.1 and Snort 1.8p1. Snort is setup to log all attempts to the logging server



The Third Level, or Secure Access level consists of a Redhat Linux 7.1 box running Tripwire 2.2.1 and IPTables and has 3 interfaces. Interface 1 is connected to the Cisco Catalyst 3500 Series XL Switch which is incoming from the Internet, the First level and the Second Level. Interface 2 is connected to the Cisco Catalyst 3500 Series XL Switch which is connected to the segment that contains the Secure Access file and print servers and an Intrusion Detection System. Interface 3 is connected to another Cisco Catalyst 3500 Series XL Switch which proceeds further into the network.

This segment starts off with a stateful firewall (IPTABLES) running on a hardened OS that employs a program that verifies file integrity (Tripwire) and sends alerts if that verification fails. The firewall is rules based and can perform masquerading if needed.

Next is a Cisco Catalyst 3500 Series XL Switch to prevent packet sniffing.

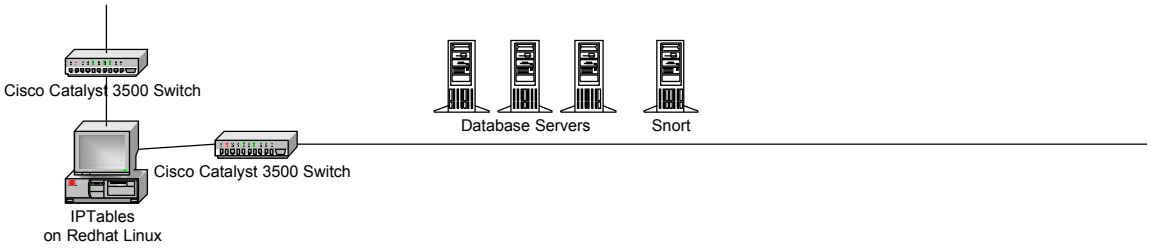
The following servers reside on this segment:

- Secure File Servers
- Print Servers
- Intrusion Detection System

These servers defined:

The File and Print Servers are running Windows 2000 SP2 using Active Directory.

The Intrusion Detection System is a hardened RedHat Linux 7.1 box running Tripwire 2.2.1 and Snort 1.8p1. Snort is setup to log all attempts to the logging server



The Fourth Level, or Secure Database Access level consists of a Redhat Linux 7.1 box running Tripwire 2.2.1 and IPTables and has 2 interfaces. Interface 1 is connected to the Cisco Catalyst 3500 Series XL Switch which is incoming from the Internet, the First, Second and Third Levels. Interface 2 is connected to the Cisco Catalyst 3500 Series XL Switch which is connected to the segment that contains the Secure Database Access Servers and an Intrusion Detection System.

This segment starts off with a stateful firewall (IPTABLES) running on a hardened OS that employs a program that verifies file integrity (Tripwire) and sends alerts if that verification fails. The firewall is rules based and can perform masquerading if needed.

Next is a Cisco Catalyst 3500 Series XL Switch to prevent packet sniffing.

The following servers reside on this segment:

- Database Servers
- Intrusion Detection System

These servers defined:

The Database Servers are running Windows 2000 SP2 using SQL Server 2000 with all current patches applied.

The Intrusion Detection System is a hardened RedHat Linux 7.1 box running Tripwire 2.2.1 and Snort 1.8p1. Snort is setup to log all attempts to the logging server

Assignment 2 - Security Policy (25 Points)

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

- Border Router
- Primary Firewall
- VPN

You may also wish to include one or more internal firewalls used to implement defense in depth or to separate business functions.

By 'security policy' we mean the specific ACLs, firewall rule set, IPSec policy, etc. (as appropriate) for the specific component used in your architecture. For each component, be sure to consider internal business operations, customers, suppliers and partners. Keep in mind you are an E-Business with customers, suppliers, and partners - you MAY NOT simply block everything!

(Special note VPNs: since IPSec VPNs are still a bit flaky when it comes to implementation, that component will be graded more loosely than the border router and primary firewall. However, be sure to define whether split-horizon is implemented, key exchange parameters, the choice of AH or ESP and why. PPP-based VPNs are also fully acceptable as long as they are well defined.)

For each security policy, write a tutorial on how to implement each ACL, rule, or policy measure on your specific component. Please use screen shots, network traffic traces, firewall log information, and/or URLs to find further information as appropriate. Be certain to include the following:

1. The service or protocol addressed by the ACL or rule, and the reason these services might be considered a vulnerability.
2. Any relevant information about the behavior of the service or protocol on the network.
3. The syntax of the ACL, filter, rule, etc.
4. A description of each of the parts of the filter.
5. An explanation of how to apply the filter.
6. If the filter is order-dependent, list any rules that should precede and/or follow this filter, and why this order is important. (Note: instead of explaining order dependencies for each individual rule, you may wish to create a separate section of your practical that describes the order in which ALL of the rules should be applied, and why.)
7. Explain how to test the ACL/filter/rule.

Be certain to point out any tips, tricks, or "gotchas".

Border Router Security

1. The security policy that will be employed on the border router will consist of extended access list that, once added, will deny all other traffic except that which is implicitly allowed. But first we will turn off unnecessary services. Keeping in mind Sans Top Ten Security Threats found at

http://www.sans.org/infosecFAQ/firewall/blocking_cisco.htm

as follows:

SANS list of commonly probed and attacked ports.

1. Block "spoofed" addresses-- packets coming from outside your company sourced from internal addresses or private (RFC1918 and network 127) addresses. Also block source routed packets.
2. Login services-- telnet (23/tcp), SSH (22/tcp), FTP (21/tcp), NetBIOS (139/tcp), rlogin et al (512/tcp through 514/tcp)
3. RPC and NFS-- Portmap/rpcbind (111/tcp and 111/udp), NFS (2049/tcp and 2049/udp), lockd (4045/tcp and 4045/udp)
4. NetBIOS in Windows NT -- 135 (tcp and udp), 137 (udp), 138 (udp), 139 (tcp). Windows 2000 – earlier ports plus 445(tcp and udp)
5. X Windows -- 6000/tcp through 6255/tcp
6. Naming services-- DNS (53/udp) to all machines which are not DNS servers, DNS zone transfers (53/tcp) except from external secondaries, LDAP (389/tcp and 389/udp)
7. Mail-- SMTP (25/tcp) to all machines, which are not external mail relays, POP (109/tcp and 110/tcp), IMAP (143/tcp)
8. Web-- HTTP (80/tcp) and SSL (443/tcp) except to external Web servers, may also want to block common high-order HTTP port choices (8000/tcp, 8080/tcp, 8888/tcp, etc.)
9. "Small Services"-- ports below 20/tcp and 20/udp, time (37/tcp and 37/udp)
10. Miscellaneous-- TFTP (69/udp), finger (79/tcp), NNTP (119/tcp), NTP (123/tcp), LPD (515/tcp), syslog (514/udp), SNMP (161/tcp and 161/udp, 162/tcp and

162/udp), BGP (179/tcp), SOCKS (1080/tcp)

11. ICMP-- block incoming echo request (ping and Windows traceroute), block outgoing echo replies, time exceeded, and destination unreachable messages **except** "packet too big" messages (type 3, code 4). (This item assumes that you are willing to forego the legitimate uses of ICMP echo request in order to block some known malicious uses.)

We will start with

enable secret which is used to set the password that grants privileged administrative access
service password-encryption encrypts passwords saved in config file.
no service udp-small-servers
no service tcp-small-servers

(The following description of small servers is excerpted from the Cisco website)
The TCP small servers are:

- Echo: Echoes back whatever you type. Type the command telnet x.x.x.x echo to see.
- Chargen: Generates a stream of ASCII data. Type the command telnet x.x.x.x chargen to see.
- Discard: Throws away whatever you type. Type the command telnet x.x.x.x discard to see.
- Daytime: Returns system date and time, if correct. It is correct if you are running NTP or have set the date and time manually from the exec level. Type the command telnet x.x.x.x daytime to see.

The UDP small servers are:

- Echo: Echoes the payload of the datagram you send.
- Discard: Silently pitches the datagram you send.
- Chargen: Pitches the datagram you send and responds with a 72 character string of ASCII characters terminated with a CR+LF

no service finger will disable the finger server
no service DHCP will disable the DHCP service
no service pad will disable PAD support
no service finger will disable the finger server
no bootp server will disable the bootp server
no ip unreachables disables ip unreachable messages to be returned

no ip redirects will stop ICMP Redirect messages
no cdp run will disable the Cisco Discovery Protocol
no ip http server will disable the web server
no ip classless will assure only configured subnets receive forwarded packets
no ip domain-lookup will stop the router from resolving host names
no ip directed-broadcast will disable broadcasts to help prevent DOS attacks
no ip source route will stop source-routed frames

Now to configure our Extended Access Lists.

The extended lists can be applied to each interface in two directions, inbound and outbound. We will apply our lists inbound to save CPU cycles.

To create our extended access list, we will use the following syntax:

access-list *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log**]

Border Router
Ingress Filter

This part of the filter addresses RFC1918 and network 127 and host 0.0.0.0 and spoofing.

Access-list 120 deny ip 10.0.0.0 0.255.255.255 any
Access-list 120 deny ip 127.0.0.0 0.255.255.255 any
Access-list 120 deny ip 172.16.0.0.0.15.255.255 any
Access-list 120 deny ip 192.168.0.0 0.0.255.255 any
Access-list 120 deny ip 224.0.0.0 31.255.255.255 any
Access-list 120 deny ip 255.0.0.0 0.255.255.255 any
Access-list 120 deny ip 169.254.0.0 0.0.255.255 any
Access-list 120 deny ip 240.0.0.0 31.255.255.255 any
Access-list 120 deny ip 248.0.0.0 31.255.255.255 any
Access-list 120 deny ip host 0.0.0.0 any
Access-list 120 deny ip x.x.34.0 0.255.255.255 any

This section blocks what I call nuisance traffic. For example, NetBIOS, Microsoft's directory services and all ICMP traffic etc.

Access-list 120 deny tcp any any range 135 139
Access-list 120 deny udp any any range 135 139
Access-list 120 deny tcp any any eq 445

Access-list 120 deny udp any any eq 445
Access-list 120 deny tcp any any eq sunrpc
Access-list 120 deny udp any any eq sunrpc
Access-list 120 deny tcp any any eq 2049
Access-list 120 deny udp any any eq 2049
Access-list 120 deny icmp any any
Access-list 120 deny tcp any any range ftp telnet
Access-list 120 deny any any range exec lpd
Access-list 120 deny tcp any any range 6000 6255

The following will provide access to remaining services.

Access-list 120 permit tcp any any

The above access list is then applied to the Serial port of the border router.
interface seriel0
ip address x.x.34.58 255.255.255.0
ip access-group 120 in

Egress Filter

This part of the filter addresses RFC1918 and network 127 and host 0.0.0.0 and allowed outbound access.

Access-list 130 permit ip x.x.34.0 0.255.255.255 any
Access-list 130 deny ip 10.0.0.0 0.255.255.255 any
Access-list 130 deny ip 127.0.0.0 0.255.255.255 any
Access-list 130 deny ip 172.16.0.0.0.15.255.255 any
Access-list 130 deny ip 192.168.0.0 0.0.255.255 any
Access-list 130 deny ip 224.0.0.0 31.255.255.255 any
Access-list 130 deny ip 255.0.0.0 0.255.255.255 any
Access-list 130 deny ip 169.254.0.0 0.0.255.255 any
Access-list 130 deny ip 240.0.0.0 31.255.255.255 any
Access-list 130 deny ip 248.0.0.0 31.255.255.255 any
Access-list 130 deny ip host 0.0.0.0 any
Access-list 130 deny icmp any any
Access-list 130 permit tcp any any established

The above access list is then applied to the Ethernet port of the border router.

interface Ethernet0
ip address x.x.34.59 255.255.255.0
ip access-group 130 in

Note: Any access out (ie: the proxy, DNS servers, SMTP etc) will use NAT.

Firewall Policy

Our exterior firewalls are Nokia IP 440 with Checkpoints Firewall 1 and VPN. The reason we have 2 is this gives us a migration path to redundant access of the Ecommerce sites in the future by employing VRRP for failover, and possibly BGP for redundant ISP's. And they currently enhance the Internet connections for the VPNs and Web access that are used by our partners and remote employees. We will also use Network Address Translation for services that will be accessed from the Internet. The base configuration of the firewalls properties will be the same on both however the rulebase will differ. This setup policy will consist of :

Firewall at x.x.34.60

Apply Rules to Interface Direction Eitherbound

Accept UDP replies

Drop all ICMP

Log implied rules

Firewall A

Firewall A is used for access to the general services offered by GIAC. Access to the General Use Web Server, the DNS Servers, and the SMTP server. This access is done by using Network Address Translation, and rules determining the specific ports accessible.

Firewall A Objects

Name	IP Address	Net Mask	NAT	Comment

Fw	x.x.34.61	255.255.255.0		Firewall
www1	10.0.30.5	255.0.0.0	x.x.34.65	General Webserver
DNS1	10.0.30.15	255.0.0.0	x.x.34.67	Exteral DNS1
DNS2	10.0.30.20	255.0.0.0	x.x.34.68	Exteral DNS2
SMTP	10.0.30.30	255.0.0.0	x.x.34.69	SMTP Server
Proxy	10.0.35.30	255.0.0.0	x.x.34.70	Web Proxy
Time	10.0.35.25	255.0.0.0	x.x.34.65	Time Server

Firewall A Rulebase

No	Source	Destination	Service	Action	Track	Comment
	Any	FW	Any	Drop	Long	To Firewall
	Any	WWW1	HTTP	Accept	Long	WebTraffic
	Any	DNS1 DNS2	Dns Tcp Dns Udp	Accept	Long	DNS
	Any	SMTP	Smtp	Accept	Long	SMTP In
	SMTP	Any	Any	Accept	Long	SMTP Out
	PROXY	ANY	HTTP HTTPS	Accept	Long	Web Browsing
	Any	Any	Any	Drop	Long	Catch all

The following explins the rulebase applied here.

Rule 1 – Any packet addressed to the firewall will be dropped and logged.
Rule 2 – Any traffic to the web server on port 80 is accepted and logged.
Rule 3 – Any traffic to the DNS servers on port 53 TCP & UDP is accepted and logged.
Rule 4 – Any traffic to the SMTP server on port 25 is accepted and logged.
Rule 5 – Any traffic from the SMTP server is accepted and logged.
Rule 6 – Any traffic from the proxy server on port 80 & 443 is accepted and logged.
Rule 7 – Any traffic to anywhere on any port is dropped and logged.

The rules are checked from top down until a match is found. The first rule is meant to stop any activity directed at the firewall and not return any information. This is generally called a stealth rule. All rules are logged. This can lead to sizable log files but seeing as we

have an abundance of disk space, having the logs available for forensic investigation can be a great help in defining how a compromise occurred.

Firewall B and VPN

Firewall B is used for VPN access, remote access and access to the Secure Web Server that contains the scripts that access the Database servers. This firewall will house our encryption domain. Our remote users will use securemote. The setup of the encryption domain and securemote is explained in further detail at:
http://support.checkpoint.com/kb/docs/public/securemote/3_0/pdf/securemote30.pdf

Our configuration will consist of the following:

We create manual IPSec key(s)

1. In the Policy Editor, select Manage/Keys. Click on New and select SPI. The Manual IPSec window is displayed.
2. In SPI Value enter a unique identifier in hex, typically between 0x100 and 0xffff.
3. Check ESP and AH.
4. Check AH; set Authentication Algorithm to MD5.
5. Select Keys/By Seed, enter some random text in the Seed box, click Generate. Write down the Encryption Key (Hex value) and Authentication Key (Hex value).
6. Select Keys/Manually and reenter the keys just generated. Click OK.
7. Repeat the New/SPI process if you want separate inbound and outbound keys.

Create network objects

In the Policy Editor, select Manage/Network objects.

A) Create Network objects for the LANs you will connect GIACHQ-lan (10.0.0.0)and aquired-lan (10.1.0.0).

B) Create a Workstation object for the aquired company's VPN. Give it a name (e.g. GIACsub1) and an IP address , comment, Location is External, Type is Gateway.

C) Click on the VPN tab. Under Domain, click Other and select remote-lan.

D) Under Encryption schemes, check Manual IPSEC.

E) Now select the Workstation object of your firewall and click Edit. Click the VPN tab. Under Domain, click Other and select HQ-lan.

F) Under Encryption scheme, check Manual IPSEC. Click OK.

Adding encryption rules

A) Add a rule for inbound traffic. Source is aquired-lan, Destination is GIACHQ-lan, Action is Encrypt.

B) Double-click on Encrypt in the Action column to display Encryption Properties.

C) Select Manual IPSec, Click on Edit.

D) Select the spi you're going to use for inbound, and select GIACsub1 as the Allowed Peer Gateway. Click OK.

E) Now add a rule for outbound traffic. Source is GIACHQ-lan,, Destination is aquired-lan, Action is Encrypt.

F) Double-click on Encrypt in the Action column to display Encryption Properties.

G) Select Manual IPSec, Click on Edit.

H) Select the spi you're going to use for outbound, and select GIACsub1 as the Allowed Peer Gateway. Click OK.

Now to add translation rules

You must add translation rules at the top of your translation rulebase so that encrypted traffic won't get NATed instead of encrypted.

1. In the Policy editor, select the Translation tab.

2. Select Edit/Add Rule at Top from the menu.
3. Set Source to GIACHQ-LAN, Destination to aquired-lan, Service to Any; leave all 3 translation entries as Original.
4. Select Edit/Add Rule at Top from the menu.
5. Set Source to aquired-lan, Dest to GIACHQ-LAN, Service to Any; leave all 3 translation entries as Original.

Firewall B Objects

Name	IP Address	Net Mask	NAT	Comment
Fw	x.x.34.60	255.255.255.0		Firewall
GIACHQ-lan	10.0.0.0			HQ Lan
aquired-lan,	10.1.0.0			Subsidiary Lan
GIACsub1	x.x.35.20			Subsidiary FW
www2	10.0.30.10	255.0.0.0	x.x.34.66	SecureWebserver
Partner	y.y.50.25	255.255.0.0		VPN Access
Remoteuser	10.0.30.20	255.0.0.0	x.x.34.68	Securemote Access

Firewall B Rulebase

No	Source	Destination	Service	Action	Track	Comment
	aquired-lan,	GIACHQ-lan	Encrypt	Accept	Long	From Subsidiary
	GIACHQ-lan	aquired-lan,	Encrypt	Accept	Long	To Subsidiary
	Any	FW	Any	Drop	Long	To Firewall
	Any	WWW1	HTTPS	Accept	Long	WebTraffic

	Any	DNS1 DNS2	Dns Tcp Dns Udp	Accept	Long	DNS
	Any	SMTP	Smtp	Accept	Long	SMTP In
	SMTP	Any	Any	Accept	Long	SMTP Out
	PROXY	ANY	HTTP HTTPS	Accept	Long	Web Browsing
	Any	Any	Any	Drop	Long	Catch all

The following explains the rulebase applied here.
Rule 1 – Packets from aquired-lan VPN will be accepted and logged.
Rule 2 – Packets to aquired-lans VPN will be accepted and logged.
Rule 3 – Any packet addressed to the firewall will be dropped and logged.
Rule 4 – Any traffic to the web server on port 443 is accepted and logged.
Rule 5 – Any traffic to the DNS servers on port 53 TCP & UDP is accepted and logged.
Rule 6 – Any traffic to the SMTP server on port 25 is accepted and logged.
Rule 7 – Any traffic from the SMTP server is accepted and logged.
Rule 8 – Any traffic from the proxy server on port 80 & 443 is accepted and logged.
Rule 9 – Any traffic to anywhere on any port is dropped and logged.

The rules are checked from top down until a match is found. The third rule is meant to stop any activity directed at the firewall and not return any information. This is generally called a stealth rule. All rules are logged. This can lead to sizable log files but seeing as we have an abundance of disk space, having the logs available for forensic investigation can be a great help in defining how a compromise occurred.

Note: Rules 1 and 2 use different SPI's.

Also on the Nokia Firewalls we are using Webshield for Nokia Appliances which is described further at

http://www.nokia.com/securitysolutions/network/webshield.html

(Excerpted from Nokia's website)
WebShield for Nokia Appliance tightly integrates the Nokia high-performance, purpose-built security platform and the McAfee market-leading, anti-virus software to deliver best-of-breed virus protection.

Secondary Firewalls

The secondary firewalls consist of RedHat Linux running IPTables. The operating system has been hardened using Bastille Linux 1.1. Tripwire has also been employed to help identify any compromises. All secondary firewall policies are "only allow what is needed

and deny everything else" We recognize that this is higher maintenance, but we find it more secure and find it more beneficial if an administrator occasionally works on a machine , he is more apt to see if "something just isn't right".

http://www.bastille-linux.org/
http://www.tripwire.org/

We have four of these firewalls running. All patches up to date. 3 of the machines have 3 interfaces and the fourth has 2. In each case the interfaces are named eth0, eth1, and eth2 (except for the fourth which just has 2 interfaces. In all cases eth0 will be the interface facing the Internet, eth1 will be facing the defined segment and eth2 will be facing further into the network. We start off with a rules policy to lock down each server. The following script was found at

http://www.linuxhelp.net/guides/davion/iptables-script

and I've found it to be very secure and self explanatory.

```
#!/bin/bash
#
# This is a sample firewall for ip_tables, the tool for doing firewalling
# and masquerading under the 2.3.x/2.4.x series of kernels.
#
# Be warned, this is a very restrictive set of firewall rules (and they
# should be, for proper security). Anything that you do not _specifically_
# allow is logged and dropped into /dev/null, so if you're wondering why
# something isn't working, check /var/log/messages.
#
# This is about as close as you get to a 'secure' firewall. It's nasty,
# it's harsh, and it will make your machine nearly invisible to the rest
# of the internet world. Have fun.
#
# To run this script you must 'chmod 700 iptables-script' and then execute
# it. To stop it from running, run 'iptables -F'

#Point this to your copy of ip_tables
IPT="/sbin/iptables"

#Load the module.
modprobe ip_tables

#Flush old rules, delete the firewall chain if it exists
$IPT -F
$IPT -F -t nat
```

```
$IPT -X firewall

#Setup Masquerading. Change the IP to your internal network and uncomment
#this in order to enable it.
#$IPT -A POSTROUTING -t nat -s 192.168.1.0/24 -j MASQUERADE
#$IPT -P FORWARD ACCEPT
#echo 1 > /proc/sys/net/ipv4/ip_forward

#Set up the firewall chain
$IPT -N firewall
$IPT -A firewall -j LOG --log-level info --log-prefix "Firewall:"
$IPT -A firewall -j DROP

#Accept ourselves
$IPT -A INPUT -s 127.0.0.1/32 -d 127.0.0.1/32 -j ACCEPT
#If you're using IP Masquerading, change this IP to whatever your internal
#IP address is and uncomment it
#$IPT -A INPUT -s 192.168.1.1/32 -d 0/0 -j ACCEPT

#Accept DNS, 'cause it's warm and friendly
$IPT -A INPUT -p udp --source-port 53 -j ACCEPT
$IPT -A INPUT -p tcp --source-port 113 -j ACCEPT
$IPT -A INPUT -p tcp --destination-port 113 -j ACCEPT

# for proxy

#$IPT -A INPUT -p tcp --destination-port 80 -j ACCEPT

#Allow ftp to send data back and forth.
#$IPT -A INPUT -p tcp ! --syn --source-port 20 --destination-port 1024:65535 -j ACC
#EPT

#Accept SSH. Duh.
$IPT -A INPUT -p tcp --destination-port 22  -j ACCEPT

#Send everything else ot the firewall.
$IPT -A INPUT -p icmp -j firewall
$IPT -A INPUT -p tcp --syn -j firewall
$IPT -A INPUT -p udp -j firewall
```

Now in our first secondary firewall which needs access to our service network we have added;

```
$IPT -A INPUT -i eth+ -p tcp –destination 10.0.30.5 -port 80 -j ACCEPT
```

(for Http)
$IPT -A INPUT -i eth+ -p tcp –destination 10.0.30.5 -port 443 -j ACCEPT
(for SSL)
$IPT -A INPUT -i eth+ -p tcp –destination 10.0.30.10 -port 80 -j ACCEPT
(for Http)
$IPT -A INPUT -i eth+ -p tcp –destination 10.0.30.10 -port 443 -j ACCEPT
(for SSL)
$IPT -A INPUT -i eth0 -p tcp –destination 10.0.30.30 -port 25 -j ACCEPT
(for SMTP)
$IPT -A INPUT -i eth1 -p tcp –destination 10.0.35.20 -port 1352 -j ACCEPT
(for Lotus Notes to communicate with SMTP server)
$IPT -A INPUT -i eth+ -p udp –destination 10.0.35.25 -port 514 -j ACCEPT
(access to SYSLOGD server)
$IPT -A INPUT -i eth2 -p tcp –s 10.0.35.30 -j ACCEPT
(access from Proxy server)
$IPT -A INPUT -i eth2 -p tcp –s 10.0.35.25 -port 123 -j ACCEPT
(allow access to external time server)
$IPT -A INPUT -i eth1 –source 10.0.30.10 –destination 10.0.45.0/24 -j ACCEPT
(for Web Server 2 in Service network to access Database servers)

Our second secondary firewall needs access for the mail server to the service network and the proxy server to the Internet and DNS.

$IPT -A INPUT -i eth1 -p tcp –destination 10.0.30.30 -port 1352 -j ACCEPT
(for Lotus Notes Mail Server to communicate with SMTP server)
$IPT -A INPUT -i eth1 -p tcp –s 10.0.35.25 -port 123 -j ACCEPT
(allow access to external time server)
$IPT -A INPUT -i eth1 -p udp –destination 10.0.35.25 -port 514 -j ACCEPT
(access to SYSLOGD server)
$IPT -A INPUT -i eth1 -p tcp –s 10.0.35.30 -j ACCEPT
(access from Proxy server)
$IPT -A INPUT -i eth1 -p tcp –destination 10.0.30.5 -port 80 -j ACCEPT
(for HTTP to Web Server in Service network)
$IPT -A INPUT -i eth1 -p tcp –destination 10.0.30.5 -port 443 -j ACCEPT
(for SSL to Web Server in Service network)
$IPT -A INPUT -i eth1 -p tcp –destination 10.0.30.10 -port 80 -j ACCEPT
(for HTTP to Web Server 2 in Service network)
$IPT -A INPUT -i eth1 -p tcp –destination 10.0.30.10 -port 443 -j ACCEPT
(for SSL to Web Server 2 in Service network)
$IPT -A INPUT -i eth0 –source 10.0.30.10 –destination 10.0.45.0/24 -j ACCEPT
(for Web Server 2 in Service network to access Database servers)
$IPT -A INPUT -i eth1 –source 10.0.35.120 –destination 10.0.40.0/24 -j ACCEPT
(for CEO to access Secure servers)
Our Third secondary firewall needs access to the secure servers. As this is done on a need to access basis, a rule will be made to access the servers using 10.0.35.120 as the CEO's

IPAddress for an example.

$IPT -A INPUT -i eth0 –source 10.0.35.120 –destination 10.0.40.0/24 -j ACCEPT
(for CEO to access Secure servers)
$IPT -A INPUT -i eth0 –source 10.0.30.10 –destination 10.0.45.0/24 -j ACCEPT
(for Web Server 2 in Service network to access Database servers)
$IPT -A INPUT -i eth1 -p udp –destination 10.0.35.25 -port 514 -j ACCEPT
(access to SYSLOGD server)

Our last secondary firewall needs access to the Database servers

$IPT -A INPUT -i eth0 –source 10.0.30.10 –destination 10.0.45.0/24 -j ACCEPT
(for Web Server 2 in Service network to access Database servers)
$IPT -A INPUT -i eth1 -p udp –destination 10.0.35.25 -port 514 -j ACCEPT
(access to SYSLOGD server)

VPN and Remote Access

Our VPN is a Nokia Firewall/VPN. The configuration is stated above in the Firewall B section.

(the following was excerpterd from http://www.nokia.com/vpn/application.html)

Nokia Firewall/VPN solutions combine best-in-class firewall capabilities with highly secure encryption and authentication features. This is achieved by integrating the proven Nokia networking platform with market-leading Check Point firewall and encryption modules. The Nokia Firewall/VPN appliance encrypts sensitive data and creates a secure tunnel between each site, thus enabling independent corporate LAN's to be connected through the Internet. The Nokia Firewall/VPN gateway performs the encryption on behalf of the LAN or host groups behind it. All of this VPN functionality is completely transparent to end users, and all existing applications are supported.

Our users Remote access is Securemote.

A description of Securemote from

http://www.checkpoint.com/products/vpn1/securemoteds.html

For remote access we are using VPN-1 SecuRemote™, Using VPN-1 SecuRemote, remote users can connect to their corporate gateways via Internet connections and establish secure VPN sessions to access sensitive network resources. When installed on LAN clients, VPN-1 SecuRemote establishes "Intranet VPN" connections to either critical application servers or internal VPN gateways. Whether internal or remote access, the VPN client transparently encrypts and authenticates critical data to protect against eavesdropping and malicious data tampering.

Assignment 3 – Audit Your Security Architecture (25 Points)

You have been assigned to provide technical support for a comprehensive information systems audit for GIAC Enterprises. You are required to audit the Primary Firewall described in Assignments 1 and 2. Your assignment is to:

1. Plan the assessment. Describe the technical appoach you recommend to assess your perimeter. Be certain to include considerations such as what shift or day you would do the assessment. Estimate costs and level of effort. Identify risks and considerations.

2. Implement the assessment. Validate that the Primary Firewall is actually implementing the security policy. Be certain to state exactly how you do this, including the tools and commands used. Include screen shots in your report if possible.

3. Conduct a perimeter analysis. Based on your assessment (and referring to data from your assessment), analyze the perimeter defense and make recommendations for improvements or alternate architectures. Diagrams are strongly recommended for this part of the assignment.

Note: DO NOT simply submit the output of nmap or a similar tool here. It is fine to use any assessment tool you choose, but annotate the output.

Plan the assessment

The first thing we will do to plan our security audit is meet with management and agree upon a time to do the audit. We will pick a low traffic time, preferably Friday at 5pm. All administrators of the various systems will be on hand or available for assistance with problems and/or explanations. After determining what traffic should be allowed in and what traffic should be allowed out as legitimate traffic, we will identify any other traffic as suspect and identify it's source and reason.

Our overall audit will include:

- A walkthrough of the premises to view physical access and controls
- A check of disaster recovery plans to verify the security of the plans, media etc.
- A review of user policies such as passwords, group access, etc
- A review of environmental alerts, access and policies of the server rooms
- A review of any alternate access such as modems, wireless etc.
- A review of current patch levels in servers and workstations
- Scans of all network segments

But for this assignment we will only detail our review of the primary firewall.

We will look at the Primary Firewall using the following tools:

Nessus which is a security scanner that has a Plug-in architecture with a large amount of available plugins at :
http://cgi.nessus.org/plugins/dump.php3

NMAP which is a network mapper with other useful functionalities such as OS identification.
http://www.insecure.org/nmap/index.html

Ethereal which is a free network protocol analyzer.
http://www.ethereal.com/

HPING which is a tool which enables you to send crafted packets.
http://www.kyuzz.org/antirez/hping.html

The audit will be preceded by packet captures at different times on all interfaces. Examination of these captures will help define expected traffic patterns and anomalies which can be researched and defined. This will help benchmark expected traffic.

We will then test the Firewall interfaces using the above tools.

Our costs to perform these tests will be mostly time based as the above software is free and there will be no downtime for the Ecommerce site. Effort to perform the audit will be where most of the investment will be. Analyzing the packet captures and benchmarking the traffic will require a lot of effort.

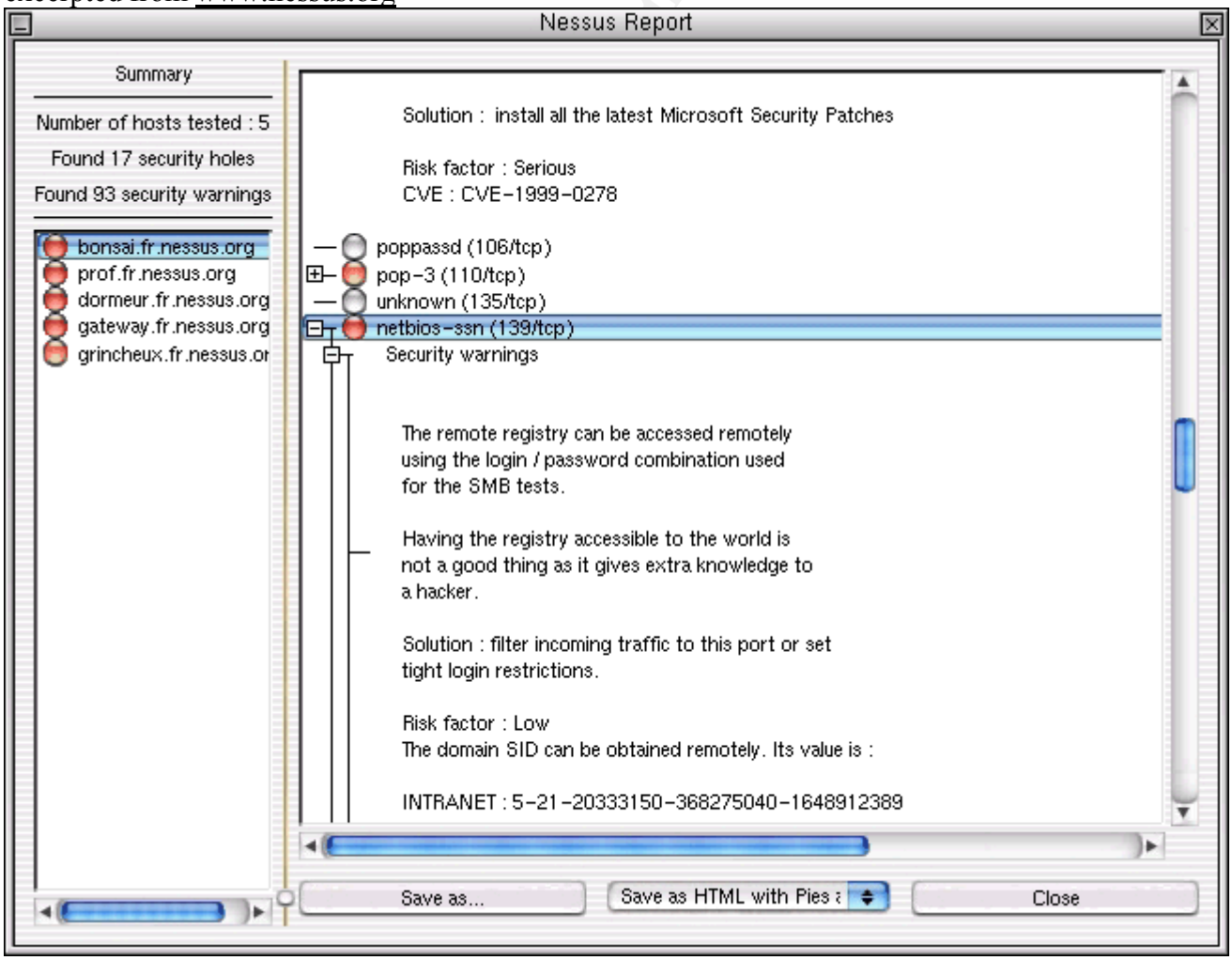
Implement the assessment.
Our first task in implementing the assessment is to preform a packet capture on all Primary Firewall interfaces using Ethereal. Taking this data we will analyze the packets and attempt to classify the data. This is a long prcess and should be done before the actual audit to supply us with a benchmark of the amount and types of traffic present. And

possibly identify any suspicious activity.

Our first task will be to identify the traffic going into and out of our network. Doing this at a packet level should give us more reconnaissance information. We will identify the expected traffic and proceed to test the firewall rulebase on a rule by rule basis, testing the expected traffic as well as explicitly denied traffic. This will be done on all interfaces.

Next we will proceed using Nessus to look for vulnerabilities using the current plug-ins. We have set up Nessus on a laptop running Redhat Linux with kernel 2.4.
We will use this laptop to perform vulnerability testing from outside the firewall on all the Ipaddresses inside the firewall, including the firewall itself. All security checks will be performed, except the Denial of Service attacks.
A list of all the Plugins available is at:
http://cgi.nessus.org/plugins/dump.php3

After Nessus is done running it will return a page similar to the following which was excerpted from www.nessus.org



which includes the vulnerability, the description, the solution and risk factor. As well as:
List of open ports
Information found on ports and descriptions.
Vulnerability found on port and where to get further information.
Similar to the following:

Information found on port netbios-ssn (139/tcp)

> The domain SID could be used to enumerate the names of the users
> in the domain.
> (we only enumerated users name whose ID is between 1000 and 1050
> for performance reasons)
> This gives extra knowledge to a cracker, which
> is not a good thing :
> - Administrator account name : Administrator (id 500)
> - Guest account name : guest (id 501)
> - BONSAI$ (id 1000)
> - IUSR_BONSAI (id 1001)
> - Renaud (id 1002)
> - thibault (id 1003)
> - MTS Trusted Impersonators (id 1005)
> - IWAM_BONSAI (id 1006)
> - Cert Requesters (id 1007)
> - Cert Server Admins (id 1008)
> - PROFWINDOWS$ (id 1009)
>
> Risk factor : Medium
> Solution : filter incoming connections to port 139

With a list of all the found vulnerabilities on the Hosts behind the firewall, we will now concentrate primarily on the firewall.

We will then proceed to use NMAP on all 4 interfaces of the firewalls to see what responses are returned. Even though we used NMAP through Nessus, I feel it is important to doublecheck the firewall itself using NMAP on it's own to perform the following scans.

The first scan we will use on all interfaces
nmap -sS -P0 –p 1-65535 -v x.x.34.60 for Firewall A
nmap -sS -P0 –p 1-65535 -v x.x.34.61 for Firewall B

The switches are explained below.
(excerpted from http://www.nmap.org/nmap/nmap_manpage.html)

-sS TCP SYN scan: This technique is often referred to as "half-open" scanning, because you don't open a full TCP connection. You send a SYN packet, as if you are going to open a real connection and you wait for a response. A SYN|ACK indicates the port is listening. A RST is indicative of a non-listener. If a SYN|ACK is received, a RST is immediately sent to tear down the connection (actually our OS kernel does this for us). The primary advantage to this scanning technique is that fewer sites will log it.

-P0 Do not try and ping hosts at all before scanning them. This allows the scanning of networks that don't allow ICMP echo requests (or responses) through their firewall. microsoft.com is an example of such a network, and thus you should always use **-P0** or **-PT80** when portscanning microsoft.com.

-v Verbose mode. This is a highly recommended option and it gives out more information about what is going on. You can use it twice for greater effect. Use **-d** a couple of times if you really want to get crazy with scrolling the screen!

The second scan we will use on all interfaces
nmap -sF -P0 –p 1-65535 -v x.x.34.60 for Firewall A
nmap –sF -P0 –p 1-65535 -v x.x.34.61 for Firewall B

The switches are explained below.
(excerpted from http://www.nmap.org/nmap/nmap_manpage.html)

-sF -sX -sN
Stealth FIN, Xmas Tree, or Null scan modes: There are times when even SYN scanning isn't clandestine enough. Some firewalls and packet filters watch for SYNs to restricted ports, and programs like Synlogger and Courtney are available to detect these scans. These advanced scans, on the other hand, may be able to pass through unmolested.

The idea is that closed ports are required to reply to your probe packet with an RST, while open ports must ignore the packets in question (see RFC 793 pp 64). The FIN scan uses a bare (surprise) FIN packet as the probe, while the Xmas tree scan turns on the FIN, URG, and PUSH flags. The Null scan turns off all flags. Unfortunately Microsoft (like usual) decided to completely ignore the standard and do things their own way. Thus this scan type will not work against systems running Windows95/NT. On the positive side, this is a good way to distinguish between the two platforms. If the scan finds open ports, you know the machine is not a Windows box. If a -sF,-sX,or -sN scan shows all ports closed, yet a SYN (-sS) scan shows ports being opened, you are probably looking at a Windows box. This is less useful now that nmap has proper OS detection built in. There are also a few other systems that are broken in the same way Windows is. They include Cisco, BSDI, HP/UX, MVS, and IRIX. All of the above send resets from the open ports when they should just drop the packet.

-P0 Do not try and ping hosts at all before scanning them. This allows the scanning of networks that don't allow ICMP echo requests (or responses) through their firewall. microsoft.com is an example of such a network, and thus you should always use **-P0** or **-PT80** when portscanning microsoft.com.

-v Verbose mode. This is a highly recommended option and it gives out more information about what is going on. You can use it twice for greater effect. Use **-d** a couple of times if you really want to get crazy with scrolling the screen!

The third scan we will use on all interfaces
nmap -sU-P0 –p 1-65535 -v x.x.34.60 for Firewall A
nmap -sU-P0 –p 1-65535 -v x.x.34.61 for Firewall B

The switches are explained below.

(excerpted from http://www.nmap.org/nmap/nmap_manpage.html)

-sU UDP scans: This method is used to determine which UDP (User Datagram Protocol, RFC 768) ports are open on a host. The technique is to send 0 byte udp packets to each port on the target machine. If we receive an ICMP port unreachable message, then the port is closed. Otherwise we assume it is open.

Some people think UDP scanning is pointless. I usually remind them of the recent Solaris rcpbind hole. Rpcbind can be found hiding on an undocumented UDP port somewhere above 32770. So it doesn't matter that 111 is blocked by the firewall. But can you find which of the more than 30,000 high ports it is listening on? With a UDP scanner you can! There is also the cDc Back Orifice backdoor program which hides on a configurable UDP port on Windows machines. Not to mention the many commonly vulnerable services that utilize UDP such as snmp, tftp, NFS, etc.

Unfortunately UDP scanning is sometimes painfully slow since most hosts impliment a suggestion in RFC 1812 (section 4.3.2.8) of limiting the ICMP error message rate. For example, the Linux kernel (in net/ipv4/icmp.h) limits destination unreachable message generation to 80 per 4 seconds, with a 1/4 second penalty if that is exceeded. Solaris has much more strict limits (about 2 messages per second) and thus takes even longer to scan. *nmap* detects this rate limiting and slows down accordingly, rather than flood the network with useless packets that will be ignored by the target machine.

As is typical, Microsoft ignored the suggestion of the RFC and does not seem to do any rate limiting at all on Win95 and NT machines. Thus we can scan all 65K ports of a Windows machine **very** quickly. Woop!

-P0 Do not try and ping hosts at all before scanning

them. This allows the scanning of networks that don't allow ICMP echo requests (or responses) through their firewall. microsoft.com is an example of such a network, and thus you should always use **-P0** or **-PT80** when portscanning microsoft.com.

-v Verbose mode. This is a highly recommended option and it gives out more information about what is going on. You can use it twice for greater effect. Use **-d** a couple of times if you really want to get crazy with scrolling the screen!

The output from these scans will help to define what ports on our Firewalls are responding to what types of packets. Below is a sample.

```
[root@o /root]# nmap -sS -P0 -vv x.x.34.61

Starting nmap V. 2.53 by fyodor@insecure.org ( www.insecure.org/nmap/ )
Interesting ports on  (x.x.34.61):
(The 297 ports scanned but not shown below are in state: filtered)
Port       State       Service
53/tcp     closed      domain
264/tcp    open        bgmp
265/tcp    open        unknown

Nmap run completed -- 1 IP address (1 host up) scanned in 154 seconds
```

Then after defining all the ports contacted on the firewall that responded we will take measures to secure them.

We will then move onto HPING and ETHEREAL. Our goal with HPING will be to craft packets that will test the Firewalls rulebase and using ETHEREAL, capture the crafted packets and responses if there are any, on both sides of the Firewalls.

With HPING we can do the following:

```
[root@o hping2]# hping --help
usage: hping host [options]
  -h  --help      show this help
  -v  --version   show version
```

```
  -c  --count     packet count
  -i  --interval  wait (uX for X microseconds, for example -i u1000)
      --fast      alias for -i u10000 (10 packets for second)
  -n  --numeric   numeric output
  -q  --quiet     quiet
  -I  --interface interface name (otherwise default routing interface)
  -V  --verbose   verbose mode
  -D  --debug     debugging info
  -z  --bind      bind ctrl+z to ttl          (default to dst port)
  -Z  --unbind    unbind ctrl+z
Mode
  default mode     TCP
  -0  --rawip      RAW IP mode
  -1  --icmp       ICMP mode
  -2  --udp        UDP mode
  -9  --listen     listen mode
IP
  -a  --spoof      spoof source address
  -t  --ttl        ttl (default 64)
  -N  --id         id (default random)
  -W  --winid      use win* id byte ordering
  -r  --rel        relativize id field         (to estimate host traffic)
  -f  --frag       split packets in more frag.  (may pass weak acl)
  -x  --morefrag   set more fragments flag
  -y  --dontfrag   set dont fragment flag
  -g  --fragoff    set the fragment offset
  -m  --mtu        set virtual mtu, implies --frag if packet size > mtu
  -o  --tos        type of service (default 0x00), try --tos help
  -G  --rroute     includes RECORD_ROUTE option and display the route buffer
  -H  --ipproto    set the IP protocol field, only in RAW IP mode
ICMP
  -C  --icmptype   icmp type (default echo request)
  -K  --icmpcode   icmp code (default 0)
      --icmp-ts    Alias for --icmp --icmptype 13 (ICMP timestamp)
      --icmp-help  display help for others icmp options
UDP/TCP
  -s  --baseport   base source port            (default random)
  -p  --destport   [+][+]<port> destination port(default 0) ctrl+z inc/dec
  -k  --keep       keep still source port
  -w  --win        winsize (default 64)
  -O  --tcpoff     set fake tcp data offset     (instead of tcphdrlen / 4)
  -Q  --seqnum     shows only tcp sequence number
  -b  --badcksum   (try to) send packets with a bad IP checksum
                   many systems will fix the IP checksum sending the packet
                   so you'll get bad UDP/TCP checksum instead.
```

```
  -M  --setseq     set TCP sequence number
  -L  --setack     set TCP ack
  -F  --fin        set FIN flag
  -S  --syn        set SYN flag
  -R  --rst        set RST flag
  -P  --push       set PUSH flag
  -A  --ack        set ACK flag
  -U  --urg        set URG flag
  -X  --xmas       set X unused flag (0x40)
  -Y  --ymas       set Y unused flag (0x80)
  --tcpexitcode    use last tcp->th_flags as exit code
  --tcp-timestamp  enable the TCP timestamp option to guess the HZ/uptime
Common
  -d  --data       data size                  (default is 0)
  -E  --file       data from file
  -e  --sign       add 'signature'
  -j  --dump       dump packets in hex
  -J  --print      dump printable characters
  -B  --safe       enable 'safe' protocol
  -u  --end        tell you when --file reached EOF and prevent rewind
  -T  --traceroute traceroute mode            (implies --bind and --ttl 1)
  --tr-keep-ttl    Keep the source TTL fixed, useful to monitor just one hop
```

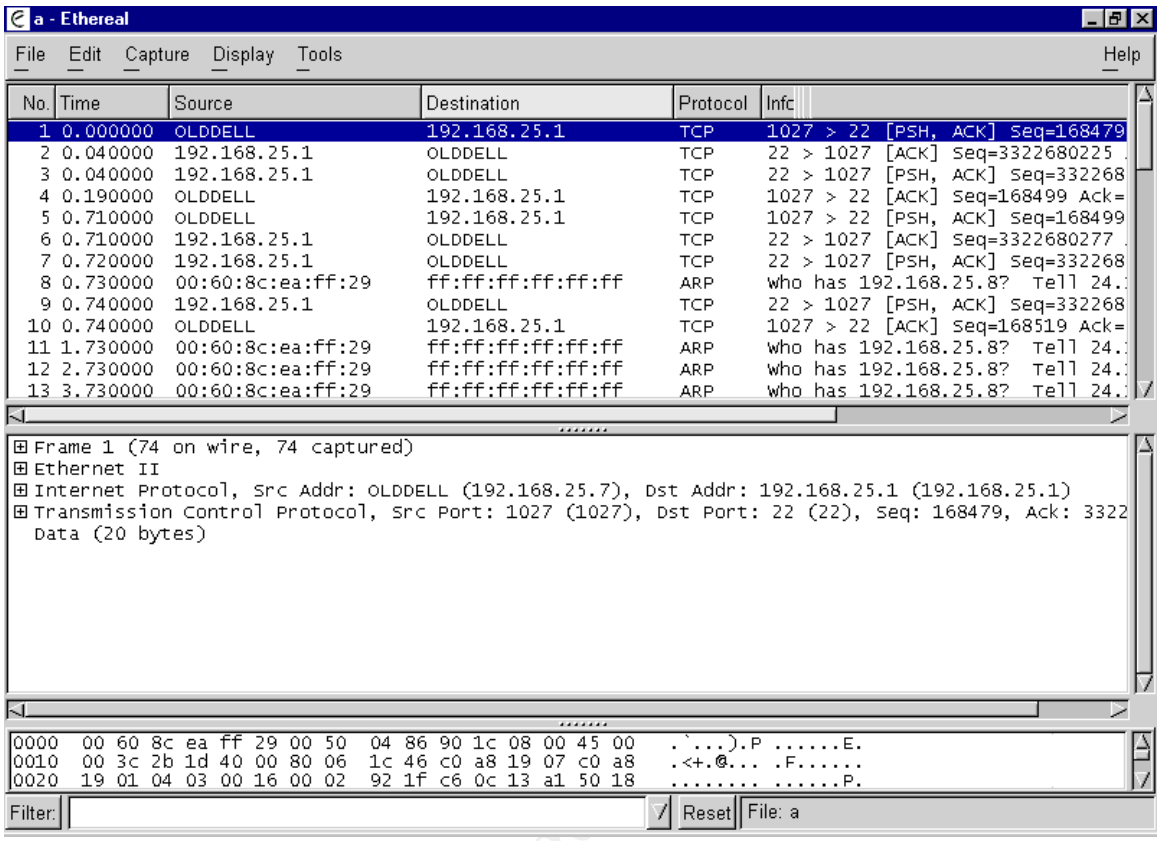
We will use the following HPING commands for a basic test.

hping –c 20 -V -S -p 80 x.x.34.6x

This will send 20 packets with the Syn flag set to port 80 of the Firewall which should not be accepting any traffic at all, so we should get something similar to the following.

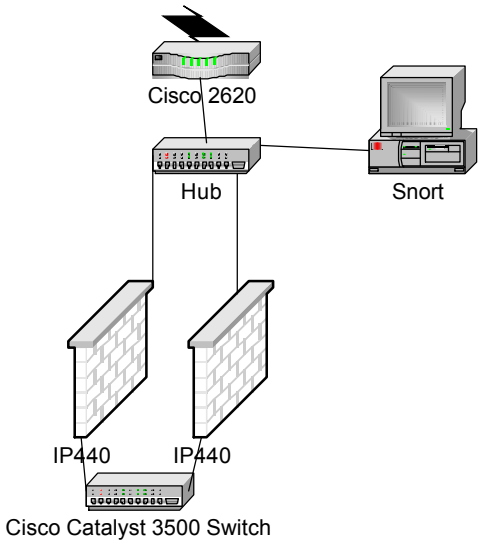
```
[root@o hping2]# hping -c 20 -V -S -p 80 x.x.34.60
eth0 default routing interface selected (according to /proc)
using eth0, addr: x.x.102.46, MTU: 1500
HPING x.x.34.60 (eth0 x.x.34.60): S set, 40 headers + 0 data bytes

--- x.x.34.60 hping statistic ---
20 packets tramitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
[root@o hping2]#
```

As we are also running ETHEREAL to verify the traffic,
This will be accomplished by using Ethereal, and will be done on each interface.



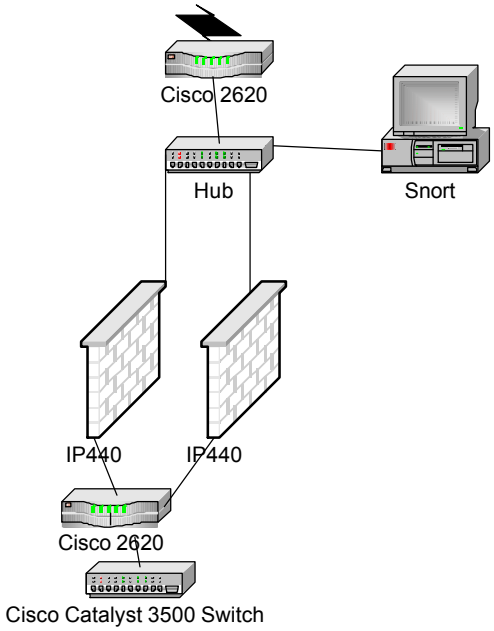
This will be done on every interface to verify the traffic that gets through is the traffic that is supposed to.

Conduct a perimeter analysis.



The first step of our perimeter analysis would be to run general reconnasaince by going to http://www,sampade.org and gathering the available information and seeing if it could be used in a compromise.

Then our perimeter analysis will also consist of taking the packet captures that we did earler and identifying the incoming traffic that is not approved and then suggest blocking it at the border router. Also if there is traffic coming from the internal network that is has not been approved we will first try to identify it's source and decide if it will be approved and we will accept it as legitmate traffic. However if it isn't approved we may want to think about adding a router to the internal network just before the Firewalls and use the ACL's to block that traffic. Or have it blocked at the firewall.



We would also use NMAP from outside to run a scan on our border router and identify

any access that may need to be closed.

Additional Recommendations would include:

Having 2 separate ISP's, preferably 2 T3's giving us a redundant Internet connection using
VRRP .

Adding an additional SMTP server and MX record of a slightly higher priority.

Create a Cluster of the Web servers for fault tolerance.

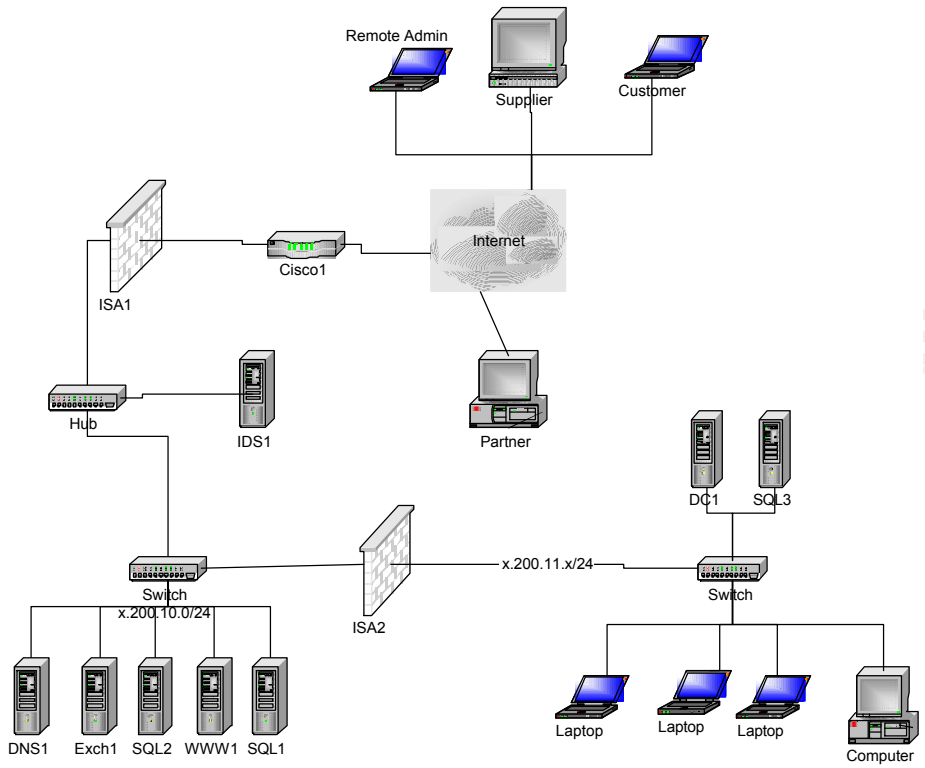
The purpose of this exercise is to help you think about threats to your network and therefore develop a more robust design. Keep in mind that the next certification group will be attacking your architecture!

Select a network design from any previously posted GCFW practicial (http://www.sans.org/giactc/gcfw.htm) and paste the graphic into your submission. Be certain to list the URL of the practical you are using. Design the following three attacks against the architecture:

1. An attack against the firewall itself. Research vulnerabilities that have been found for the type of firewall chosen for the design. Choose an attack and explain the results of running that attack against the firewall.
2. A denial of service attack. Subject the design to a theoretical attack from 50 compromised cable modem/DSL systems using TCP SYN, UDP, or ICMP floods. Describe the coutermeasures that can be put into place to mitigate the attack that you chose.
3. An attack plan to compromise an internal system through the perimeter system. Select a target, explain your reasons for choosing that target, and describe the process to compromise the target.

Note: this is the second time this assignment has been used. The first time, a number of students came up with magical "hand-waving" attacks. You must supply documentation (preferably a URL) for any vulnerability you use in your attack, and the exploit code that you use to accomplish the attack. The purpose of this exercise is for the student to clearly demonstrate they understand that firewall and perimeter systems are not magic "silver bullets" immune to all attacks.

The following is an excert from
http://www.sans.org/y2k/practical/Paul_Young_GCFW.zip



1. Cisco 2600 Router – (Cisco1)

   This router will for the first line of defence. As this is a device that deals in IP it is sensible at this point to remove any obvious IP problems.
   The router is to pass all traffic and only BLOCK the following

   a. Any packet with a source address of internal IP ranges inbound on the external interface
   b. Any packet with a source address of reserved IP ranges inbound on the external interface
   c. Any packet with a destination address of an IP Broadcast range inbound on the external interface
   d. Any packets inbound or outbound to the NetBios Port 135-139

   Although the Netbios could be blocked at the firewall effectively, it is safer to block it here. These requests are constant and fill firewall log files easily. Also the firewall is a MS device and possibly subject to this problem if accidently misconfigured. Blocking NetBios at the router solves all these problems.

2. External Firewall – MS ISA Server on Windows 2000 Server – (ISA1)

   The ISA Firewall provides multiple solutions in one box without compromising security. There is support for the following items on this device

   a. Packet level stateful firewall between internal and external interfaces
   b. HTTP Proxy support with Mime filtering
   c. VPN support using IPSec or PPTP

d. Full logging capabilities to a remote SQL server
e. Ease of management and monitoring – improved reliability and reduced chance of misconfiguration
f. HTTP Caching to reduce network load
g. Intrusion detection for common attacks – with alerting capabilites

This device it to be configured to control what data comes to and from the Internet, and to where it goes. It is also to function as a termination point for remote clients using VPN's to access servers in the service network. It will ensure that the only systems available to the outside world are those that we specify.

3. Internal Firewall – MS ISA Server on Windows 2000 Server – (ISA2)

This ISA firewall is to be configured to only allow limited access between the internal and service networks. A second firewall has been chosen rather than a 3 pronged approach for 2 reasons.

a. If the external firewall is compromised the internal network has a greater chance of remaining secure
b. ISA server only support packet filtering between Internal and External interfaces, not individually on each interface. It would therefore be impossible to tightly control data flow using packet filtering. Whilst proxy capabilities can be used to reduce this, it is much more secure to simply separate the two servers.

This firewall is to be configured to pass only HTTP/S access to the outside world. Limited access will be available to the service network as it is regarded as untrusted. Most of the services in the internal network will replicate out to the servers in the service network.. Any data flowing back in will be subject to thorough screening and filtering. It is impossible however not to allow some data back through (eg SQL tranactions) as this is a basic business function.

4. Log File Server – SQL server on Windows 2000 - (SQL2)

Due to the size of the Log files generated by the servers in the service network, it is much easier to manage them if they are stored in a database. Automated queries can then be generated to look for abnormal behaviour, and to track usage over time. The IIS web server and External firewall are to both log all their details to the SQL Logging server. The SQL server will have to have strong permissions set to control access. It can also be configured using RRAS to only communicate with those devices that are supposed to be submitting logs and queries.

5. Web Server – IIS 5 on Windows 2000 - (WWW1)

IIS is widely used throughout the web and has proven to be a reliable web server. Although it is subject to a number of well known vulnerabilities, these can be managed. The IIS server is to provide a web based interface for database and email access. Due to the attention IIS receives it will be extremely important to ensure this server is kept up to date with security

patches. It will also be critical to ensure the database developers use secure methods of performing transactions that will filter user input data.
As this is the primary point of contact for the outside world it is vital this machine stay secure. This is not a database and it's configuration is moderately static. This makes it a perfect candidate for runnning Tripwire. Any attacks that are successful would then immediately result in alerts being generated. This is doubly necessary as attacks could come through SSL which will bypass the IDS. The SSL attacks are quite unlikely however as client certificates are required for SSL sessions.

6. Snort IDS on Windows 2000

Snort is now available with support for a Windows platform. This ensures consistency with the rest of the systems, reducing the chance of Administrators misconfiguring systems they are not familiar with. The IDS will filter all incoming traffic to look for well known attack signatures. This will have to be kept relavent with regular signature updates from the Snort IDS site. There is also the possibility of this system being overwhelmed by tools such as "Stick", however the benefits outweigh the occasional false alarm and additional maintainence.

7. MS DNS Server – (DNS1)

This server will function solely as a local DNS server to answer queries from the internet. BIND has been subject to a number of vulnerabilities over time, and lately these have been very serious. Microsoft DNS has been unaffected by these so far, however the system is to be independent despite this. If any vulnerability arises compromise will have reduced effect. This machine is to be configured as a secondary to one of the zones on the internal server. The internal zone will not be published to this server. The Primary Zone for the service network will be kept on an internal server and replicated down. This can then be further configured to replicate to a server on the ISP's location, provided they can secure their box.

8. Ethernet Switch

The implementation of a Switch to handle network traffic in the service network will help improve data security. If a system is compromised and run in Promiscuous mode to obtain further information, the only data available will be Broadcast or directed to the compromised server. This will help delay an attacker and limit information exposure until the administrators detect the attack and take steps to remove the problem. It increases the risk that the IDS will not pick up attacks, however these should be detected on their way in as all data must pass the IDS

9. MS Exchange Server / AD – (Exch1)

This system is to offer email service to internal clients. The only communication to the outside world to and from this system should be

SMTP. If external clients wish to check email it is to be through Outlook Web Access through HTTPS. Limited services will be made available from this system to the internal network (MAPI, LDAP). This system will also have to function as a Domain Controller and assist in authentiaction of clients requesting authentication to systems on the service network. Replication will have to be allowed between the Domain controllers.

10. SQL Database Server – (SQL1)

    This server is to function as the back end for any queries run against it. The permissions are to be configured so as to tighly control access to the database. RRAS can be configured to control what systems can perform queries against this server. It will need to be access directly by the partner networks however, so some risk must be taken. That's why they call this Risk Management.

11. Internal SQL Server – (SQL3)

    This server is replicated with the server in the service network. By having 2 servers it is possible to stop the internal machines performing queries on the service network. This way any abnormal traffic between the service network and the internal network will be easier to detect. This replication increases the chance of incorrect data being pulled in, however this is where the permissions on the database, and the application designers have to be cautious as to what queries they allow their applications to make. The Internal Firewall can be configured to only allow access between the two SQL servers and no other machines.

12. AD / DNS / WINS / DHCP Server - (DC1)

This machine will provide basic network services to the internal network. The DNS is to contain internal and external addresses. The external addresses are to be a separate zone that is replicated to the external DNS server. The internal zone is not to be replicated out.

An attack against the firewall itself

While looking at the architecture and doing some research, I found some interesting things that I would see if this site was vulnerable to. As the firewall is running on a Windows 2000 machine, I will attempt to compromise the OS using exploits and combination of exploits. The first being an exploit I found in a posting, shows it may be possible to run arbitrary commands on IIS 5 (Win 2000) using the following URL:

http://address.of.iis5.system/scripts/..%c1%1c../winnt/system32/cmd.exe?/c+dir+c:\

An explanation of this exploit is at
http://www.wiretrip.net/rfp/p/doc.asp?id=57&iface=5

This exploit is recognized by Microsoft and a bulletin about it can be found at

http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/MS00-078.asp?frame=true&hidetoc=true

titled:
Microsoft Security Bulletin (MS00-078) Patch Available for "Web Server Folder Traversal" Vulnerability

Note: This is mostly a Unicode exploit, that seems to work on machines that have foreign Unicode fonts, however.

Then on to a more powerful attack that can be easily overlooked. The Windows 2000 IIS 5.0 Remote buffer overflow vulnerability (Remote SYSTEM Level Access)

A full explanation of this can be found at:
http://archives.neohapsis.com/archives/ntbugtraq/2001-q2/0024.html

The following excerpt from
http://archives.neohapsis.com/archives/ntbugtraq/2001-q2/0024.html
shows the potential damage that can occur.

We would like to note that eEye Digital Security did provide Microsoft with a working example exploit that when ran against a web server would, in a matter of a few seconds, bind a cmd.exe command prompt to a port on a remote IIS 5.0 web server so that a remote attacker could then execute commands with SYSTEM level access and therefore have full control of the vulnerable machine.

A denial of service attack

To begin, we have 50 compromised machines connected via Cable modems. As most of the machines out there are Windows, I have geared my attack on that premise. First I use a scheduler to put on my compromised hosts. It is a command line utility called Delayexec. It uses the syntax "delayexec [program to run] [# of seconds to launch]. Then I also installed BMB2 which is also a command line utility that is a UDP bomber. The syntax used here is BMB2 [Target IP] [Port]. By creating a batch file that consists of BMB2's target IP and port and then running DELAYEXEC to schedule all our compromised hosts launch of BMB2 , a coordinated attack using UDP on port 80 will begin. I've made sure my hosts leave their computer running all night. The attack will begin at 3 am so there is less chance of any of my hosts to identify it thereby giving me the maximum attack capability.

The border router seems to be just blocking the following:

- Private Ranges - 10.0.0.0 / 8, 172.16.0.0 /12, 192.168.0.0 /16 - RFC1918
- Multicast (Class D) - 224.0.0.0 - 239.255.255.255
- Loopback - 127.0.0.0 /8
- Broadcast - 0.0.0.0
- The range of addresses used in the internal networks – x.200.10.0, x.200.11.0
- Ranges not publicly assigned - 169.254.0.0/16, 240.0.0.0/5 & 248.0.0.0/5

Simple but effective.

DelayExec can be found at
http://www.nonags.com/nonags/cl.html
and BMB2 can be found at
http://www.tlsecurity.net/windows/DoS/

An attack plan to compromise an internal system through the perimeter system.

Now that I am experienced with this particular vulnerability seeing as I used it against the firewall, and the network design made this somewhat easier as the Webserver is running on a Windows 2000 machine using IIS5 as well, I will attempt to compromise the OS using exploits and combination of exploits. The first being an exploit I found in a posting that it may be possible to run arbitrary commands on IIS 5 (Win 2000) using the following URL:

http://address.of.iis5.system/scripts/..%c1%1c../winnt/system32/cmd.exe?/c+dir+c:\

An explanation of this exploit is at
http://www.wiretrip.net/rfp/p/doc.asp?id=57&iface=5

This exploit is recognized by Microsoft and a bulletin about it can be found at

http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/MS00-078.asp?frame=true&hidetoc=true

titled:
Microsoft Security Bulletin (MS00-078) Patch Available for "Web Server Folder Traversal" Vulnerability

Note: This is mostly a Unicode exploit, that seems to work on machines that have foreign Unicode fonts, however.

Then on to a more powerful attack that can be easily overlooked. The Windows 2000 IIS 5.0 Remote buffer overflow vulnerability (Remote SYSTEM Level Access)

A explanation of this can be found at:
http://archives.neohapsis.com/archives/ntbugtraq/2001-q2/0024.html
and at
http://www.securityfocus.com/frames/?content=/vdb/bottom.html%3Fvid%3D2674

The following excerpt from
http://archives.neohapsis.com/archives/ntbugtraq/2001-q2/0024.html
shows the potental damage that can occur.

We would like to note that eEye Digital Security did provide Microsoft with a working example exploit that when ran against a web server would, in a matter of a few seconds, bind a cmd.exe command prompt to a port on a remote IIS 5.0 web server so that a remote attacker could then execute commands with SYSTEM level access and therefore have full control of the vulnerable machine.

Additionally, if for some reason we couldn't compromise the OS and if the site was a password protected site we could use tools like Crack Whore or Brutus to get the password. And we all know how many people use the same passwords for all their logins.

Crack Whore can be found at:
http://www.subreality.net/

Brutus can be found at
http://hoobie.net/brutus/