Global Information Assurance Certification Paper

GIAC LevelTwo

Firewalls, Perimeter Protection, and VPNs

Practical Assignment for GCFW

Capitol SANS, December 2000

Written by: Ken Dill
Submitted: 2/20/2001

Assignment 1 - Security Architecture (25 Points)

Define a security architecture for GIAC Enterprises, a new Internet startup that expects to earn $200 million per year in online sales of fortune cookie sayings. Produce a diagram or set of diagrams with explanatory text that define how to use perimeter technologies to implement your security architecture.

You must consider and define access for:

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

Your architecture must specify filtering routers, firewalls, VPNs to partners, secure remote access, and internal firewalls. Be explicit about the brand and version of each perimeter defense component.

Architectural Design Methodology for GIAC Enterprises

The process of designing a secure network to address the needs of GIAC Enterprises will define and address the following major areas of concern:

1) Assets to be protected, and services to be provided.
2) Threats to be mitigated to optimally protect specific resources, network segments, and the network as a whole.
3) Access requirements for each category of user and how that access will be kept secure.
4) Network topology that addresses needs for security and accessibility.
5) Security devices including filtering routers, firewalls, VPN, and other devices that will be required to partition the network as designed.
6) Integrity assurance procedures and devices that provide additional layers of security in the form of alerts and activity logging.

Security Architecture for GIAC Enterprises

1) Assets to be protected, and services to be provided.

The most critical data for GIAC Enterprises will be stored on its database servers. This data must be accessible to employees, customers, suppliers, and partners, thus it must be allowed to be accessed across the network while being kept as secure as possible. Also of crucial importance to GIAC Enterprises is the security of financial transactions with customers, partners and suppliers. Losing its reputation for being able to provide secure transactions could be as deadly to the company as having its product stolen. Lastly the company needs to protect its network infrastructure. Losing time and productivity due to system failure will hurt the bottom line as well, so network devices and resources that support day-to-day operations for employees will also need to be protected. Major areas of concern are detailed below:

a) *The entire network* will need to be protected from potentially crippling attacks from external as well as internal sources. Examples of such attacks include DDoS, SMURF, sniffing and spoofing (described in more detail later). Rule sets will be implemented on perimeter devices to reduce the risk of these attacks.

b) *Database servers* will need to be protected from external and internal attacks as well. Internal and external firewalls will be configured to allow only approved database communications from approved sources.

c) A r*esearch and development* network will need to be protected from all external as well as internal access. No traffic will need to be permitted inbound to this network.

d) *Data transmitted across the wire* will need to be protected from interception. Intercepted communications can not only be a way for hackers to learn company secrets, or to steal the (text-based) product, but also to learn about the structure of the network in order to plan other, more damaging attacks. On external networks, encryption and tunneling will be used. On internal networks where encryption is not used, vigilance should be exercised to restrict the existence of sniffers, especially those that can be covertly installed to legitimate computers.

e) *Authorization for access* to the network must be tightly restricted, and securely maintained. Users connecting from remote sites must have sufficiently complex password requirements, session time-outs, and encrypted communications. These requirements will be enforced at the connection to the VPN device. Physical access to internal areas must be restricted as well, especially where the more sensitive data and management workstations are located.

f) *The perimeter defense devices themselves* must be protected with appropriate access lists, password protection, and defenses against Denial of Service (DoS) attacks. Remote management of these devices needs to be limited, and encrypted wherever possible. And by all means, these devices need to be kept behind locked doors with very limited access.

g) Protecting *day-to-day operations* equipment will demand that those critical resources are protected by a firewall as well. These services include email, shared files, printing services, Internet access, and so on.

h) *The framework of perimeter defenses* (consisting of routers, firewalls, and VPN) must have a method of protecting itself, detecting potential attacks in progress, logging such activity for later analysis, alerting administrators, and/or shutting down malicious access.

2) Threats to be mitigated to optimally protect specific resources, network segments, and the network as a whole.

a) Sniffing - the process of recording or analyzing all network traffic which touches a network interface card (NIC). To be an effective sniffer, the NIC must be set enabled in "promiscuous mode," where it will listen to all traffic that touches it, not just the traffic directed to that computer's address. Hackers can install sniffers on compromised systems and use them to obtain usernames, passwords, credit card numbers, personal information, and other information that could be damaging to the company. To protect against sniffing, the following guidelines should be followed:
   i) Protect hosts from having sniffer software installed to them from hackers.
   ii) Use a switched network which will reduce the traffic reaching the NIC of any computer which has been compromised with sniffer software.
   iii) Use encryption where practical so that if traffic is collected, it will be very time consuming, if not impossible to decipher.

b) DoS attacks flood a network with an enormous amount of traffic causing network devices to become overloaded by attempting to handle or route each packet. Within minutes,

network activity rises exponentially, and the network stops responding to normal traffic and service requests. One type of DoS attack is a SYN flood. This attack takes advantage of the nature of TCP/IP session establishment in which a client sends a request for communication to a server with the SYN flag set in the packet, and the server replies with a SYN-ACK message back, acknowledging the initial SYN and waiting for the final ACK from the client before continuing the session. The weakness exploited when the server allocates memory resources to holding that potential session open while waiting for the final ACK. If too many session establishments are left open, the server can lock up, or lose TCP/IP connectivity all together. These half-open connections will eventually time out, but if too many are received in a short enough period of time, the server may crash. The firewalls in this network will act as proxies for internal hosts, and do the work of managing those half-open sessions and only pass-on legitimate connections at levels that the servers are able to handle.

c) Spoofing Spoofing is the reason the attacker does not suffer a flood of SYN-ACKs is that s/he has crafted the SYN packets so that they appear to have come from another source entirely. This method of crafting packets with a bogus source address is called spoofing. Spoofing also makes it much harder to trace the actually source of such attacks. Spoofing can also be used to impersonate actual (legitimate) clients or the hijack current sessions, or replay recorded (sniffed) communications that have already taken place.

d) DDoS is a supped-up, Distributed DoS attack that takes advantage of another common aspect of IP communications, ICMP. ICMP builds in functionality to the IP protocol stack to convey status and error information including notification of network congestion and other network-related problems or to determine if a computer on the Internet is responding. The common tool of ICMP is PING in which a host sends and echo request to a target which is expected to reply to the host with an echo-reply. And there lies the aspect of ICMP which is exploited… the attacker would again craft custom packets in which the source address would be the broadcast address so that all computers on the network could join the ICMP frenzy, eventually congesting the network and making it impossible for any host or server on the network to effectively communicate. The DDoS attacker typically gains control of numerous hosts across the Internet and launches a DoS attack on a common target from multiple points simultaneously. The attacker installs DoS software on each of these compromised systems over a period of time then sends a command to all of them at once to initiate an attack on a common target.

e) Smurf is a form of DDoS attack named for the first distributed software designed to launch such attacks. Smurf attackers craft packets that would be directed to numerous hosts on a network simultaneously (typically by being sent to the broadcast address for that network). The distinction here is that the source address in the crafted packers would be entered as the address of the real target machine. This causes all hosts on that network to simultaneously flood one target with ICMP packets.

f) LAND attacks are just one more type of DoS attack, but this one can be leveraged with sending just one packet. The packet is crafted to have a source and destination address of the interface being targeted. If the system on that interface does not have a protection in place, that packet will cause an infinite loop, causing the system to crash.

g) Network mapping is another purpose for ICMP and another protocol, “finger”. These services can be used to scope out the design and connectivity of the internet network to allow hackers to plan more elaborate attacks.

h) Domain name service poisoning can occur when a hacker plants bogus DNS updates inside valid requests to DNS servers. To make DNS name transfers more efficient, DNS query packets allow space for additional options including updated entries for the target server. A bogus entry would be read by the server and entered into the table, potentially causing it to pass out bogus information in response to subsequent, legitimate queries.

i) Bypassing routing tables can be accomplished in some networks when source routing is enabled, so it will be disabled in this network. Packets containing source routes can prompt routers to re-direct them out interfaces where they should not go. The same router that allowed the packet in would then correctly guide the reply back to the source. To avoid this risk, devices in this network will disable source routing and IP redirection.

3) Access requirements for each category of user and how that access will be kept secure.

a) Customers will need access to the company's web servers in order to browse product catalogs, and make secure purchases. Secure transactions will be supported by the use of SSL between the customer and the web servers. SSL is an standard Internet web protocol that provides encryption of data between a web server and a client even during address translation.

b) Suppliers will need to view inventory levels and upload new fortunes. Since the product for GIAC enterprises is essentially just text, suppliers will be able to use the same methods to upload their new products as customers use to download purchased ones. Suppliers will be granted login IDs and sufficiently complex passwords which they will use to login through the web servers. The web servers will pass through the actual authentication to the database proxy server that in turn will verify authentication with the actual database server. Alternatively, certain suppliers may be granted VPN access as well to allow them to connect through the VPN device similar to the method supported for remote users.

c) Partners will access the GIAC Enterprises resources via VPN as well, but that connection will be a peer-to-peer connection from the VPN device at the partner's site to the VPN device at GIAC Enterprises. Once connected, connections from partner sites will still be screened by the internal firewall. Split tunneling (the ability for a remote site/user to come in from one tunnel and communicate out through another tunnel) will not be permitted.

d) Remote users will also be able to access the company network via the remote access VPN client. The VPN solution will support remote users on several OS platforms. Users will authenticate through the VPN to a RADIUS server on the internal network, allowing potential for a single-sign on. The use of one user ID and password can simplify access as well as the termination of accounts. It can increase the risk of intrusion if a user's ID and password are discovered. For this reason, the client software will be configured to not allow passwords to be stored in the connection screen, requiring users to physically type in their password each time they want to connect.

e) Users inside the company LAN will access internal database and shared file resources from their workstations on the user network through a firewall. Internal users will also be able to access the Internet through the firewalls as well. Internal network resources such as file and print servers will be available only to internal computers, and will be protected from other internal network segments and from the Internet by an internal firewall.

4) Network topology that addresses needs for security and accessibility.

This section defines devices, layout, and device configurations that will both provide protection for the data, and allow secure access to the data as needed. The GIAC Enterprises network will seek to segment resources behind external and internal firewalls to provide a layered security model. This network will be partitioned so resources most critical to the company are the most carefully protected. The overall network will be divided into nine distinct networks in two major categories, public and private.

The public networks are those which either use public addresses, or which offer resources for the outside world, such as the company's Web and DNS servers. The DMZ will be included in the list public networks given that a range of public IP addresses will be in use there, but no traffic will be allowed inbound to that network from the Internet, nor are there resources there which will be made available to the public. The four public networks are described below:

a) The External Network will be defined as the network between the perimeter router and the external firewall. The purpose of this network is to provide a screened buffer from the Internet behind the perimeter filtering router. The filtering router provides connectivity to the Internet while it screens most unwanted traffic, reducing load on the external firewall, and reducing risk of certain types of attacks such as those which use spoofing. To address spoofing risks, the perimeter router will enforce ingress and egress filtering to deny invalid traffic based on source address. This router will also provide some protocol screening as appropriate to the other internal networks, though all access lists established at the router will be reinforced by the firewalls. The external interface or the external firewall will provide static Network Address Translation (NAT) for devices on the Service Network. In short, NAT is a process of mapping network addresses that are valid only on one side of a device to addresses that are only valid on the other side. In this case public IP addresses (valid on the Internet) are being mapped to private IP address on the Service Network. The external interface of the firewall will then act as if it has multiple IP addresses (one for itself, and one for each server or group of computers for which it provides NAT). NAT has a number of beneficial effects in terms of security:
   i) Servers are more 'invisible' to the outside since the servers' actual addresses are never seen by the outside world.
   ii) Any attacks to that actual, private, address of the server will never reach that server since there will be no route to those addresses.
   iii) The firewall intercepts attacks that are directed to the server's public IP address thus protecting the operating system (OS) and IP stack on that server.

b) The Service Network will exist off a second interface behind the external firewall, and will house the public access servers including web, mail, external DNS, and database proxy servers. These servers will be hidden behind the external firewall with static NAT. An exception to the use of NAT will be the database proxy server which will not be directly accessible from the Internet, i.e. it will not have a public NAT address. It will, however be able to exchange database-related traffic with the internal database server, as will the mail server be able to transfer SMTP messages with the internal mail server.

c) The VPN Network will exist off a third interface of the external firewall. The only device on that network will be the VPN device, thus the only traffic that will pass through that firewall interface will be encrypted/tunneled traffic. The VPN device will be configured to only allow access from approved, authenticated sources, including business partners, remote users, and some suppliers. VPN peer devices at the partner sites will authenticate directly with the VPN device at GIAC Enterprises. Remote users and suppliers will "pass-through" their authentication to the RADIUS server on the inside of the VPN device. This will provide a "single-sign-on" convenience for those users. Data transmitted to and from the external interface of the VPN will be authenticated and encrypted using IPSec and the Encapsulating Security Payload (ESP) encapsulation method. ESP offers an advantage over the popular alternative, Authentication Header (AH), in that it encrypts the original IP address, while AH only authenticates it. Secure Hash Algorithm (SHA) will be chosen here is over Message Digest 5 (MD5) as the authentication method because it provides a longer hash value – more difficult to crack. Likewise, Triple Data Encryption Standard (3DES) will be used instead of DES because it provides increased resistance to brute-force cracking techniques in that it runs the same encryption algorithm as DES, only it runs it three times. It should be noted that these methods of encryption will take additional processing power on the part of the VPN device and the clients.

d) The Internal Network will be accessible across a de-militarized zone (DMZ) off the fourth interface of the Internal network. No traffic will be permitted to initiate from any other interface on that firewall to the Internal network except for the database and mail proxy servers communicating with their corresponding primary internal servers. The only traffic allowed through that interface from the Internal Network will be necessary traffic such as HTTP, DNS, FTP, HTTPS, and SMTP.

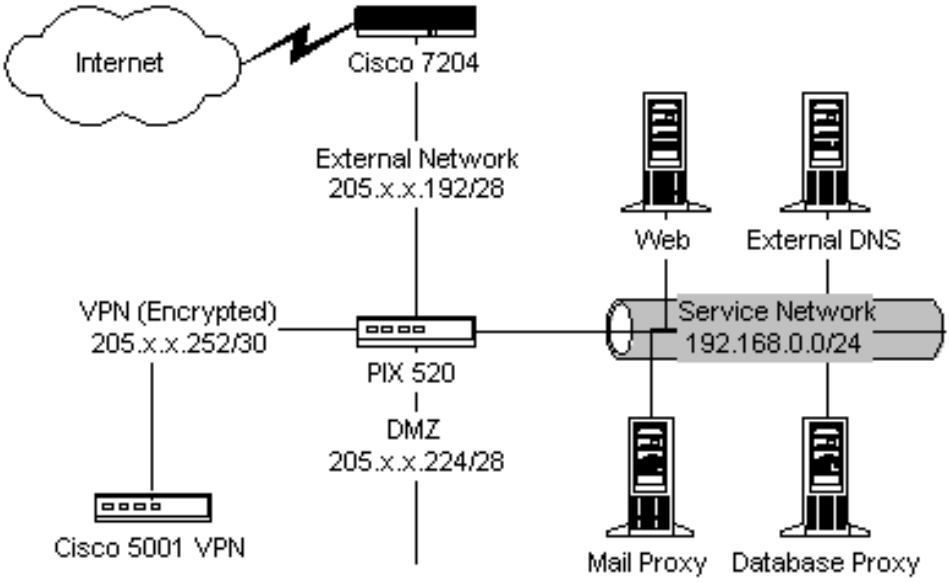


Figure 1 - The public networks behind the filtering router and external firewall.

The private networks are those which either use private address schemes, or which contain resources that are only intended for internal access. These resources include internal mail servers, file servers, and print servers, along with the users' computers themselves. Private, internal networks (not publicly accessible) will be segregated from the outside world, and each other by the use of a second, internal firewall with multiple interfaces. Items 'e' through 'i' below describe these five internal networks:

e) The VPN network will contain virtual connections from business partners and remote access users. These connections will be assigned addresses by the VPN device as they connect. The pool of addresses to be assigned will be viable private addresses consistent with the addressing scheme used for the rest of the internal networks. Virtual connections on the VPN network will only be able to access necessary services on the file server and database networks, as well as being permitted to browse to the Internet across the DMZ.

f) The user network will also have access to services on the file server and database networks and permissions to browse the Internet.

g) The File Server Network will contain the most critical day-to-day business operation systems including file, mail, intranet, and internal DNS servers. Certain devices on this network will need to be accessible from selected devices/users from all internal networks, as well as the mail proxy from the service network.

h) The database network will host only database servers, and will be accessible to all other internal networks, and the database proxy server.

i) The research and development network will deny all inbound traffic from all other networks. Hosts on that network will be permitted access to other internal resource networks and to the Internet.

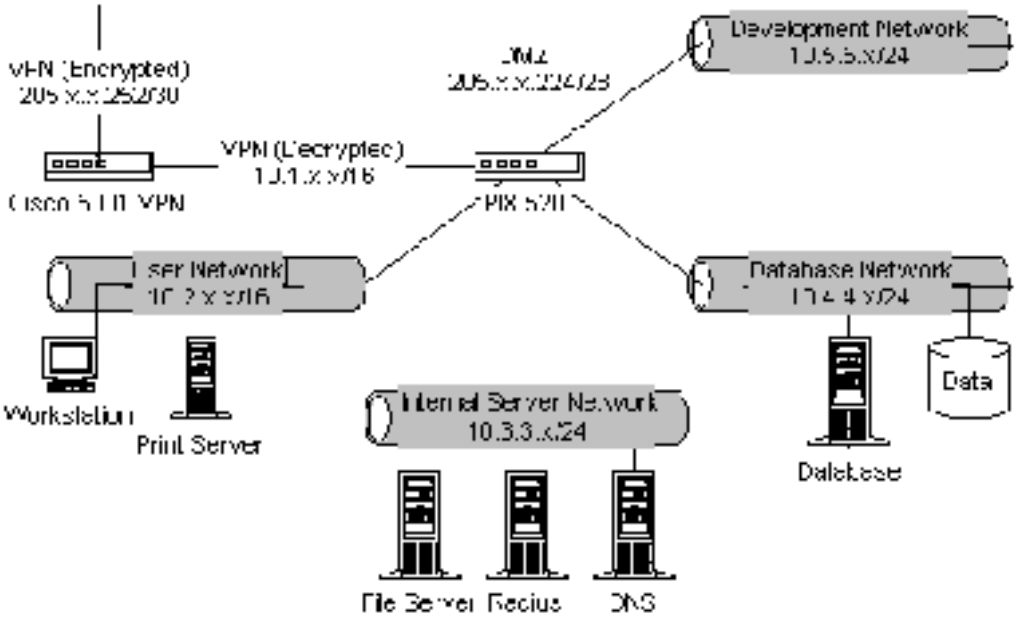


Figure 2 - The 'private' networks behind the VPN and internal firewall.

5) Security devices including filtering routers, firewalls, VPN, and other devices that will be required to partition the network as designed.

a) The Perimeter Router will be a Cisco 7204 VXR NPE-300 with IOS version 12.0(7)T, one WAN interface and one LAN interface. Redundant connectivity to the Internet through different service providers is highly recommended, as well as maintaining device redundancy. This device was chosen for the following characteristics:
   i) Extra slots will be available for redundant WAN connections and/or additional LAN interfaces for a fail-over configuration with another router
   ii) Bandwidth scalability with support for multiple WAN connections of T1, T3, or OC3
   iii) Optional support for digital voice and video coder-decoder (CODEC) connectivity
   iv) Optional support VPN gateway services
   v) Optional support of the Cisco IOS Firewall feature set
   vi) IOS 12.0(7)T is not susceptible to a recently identified DoS attack vulnerability to which other 12.x versions of the IOS are susceptible

b) The External Firewall will be a Cisco PIX 520 UR v 5.3(1) with 4 10/100 Fast Ethernet interfaces. The primary purpose of this device will be to screen access based on stateful connections and to provide Network Address Translation for publicly accessible servers on the service network. This firewall was chosen for the following characteristics:
   i) Hardware-based firewall (no worries about vulnerabilities in an underlying OS)
   ii) Up to 250,000 simultaneous connections
   iii) Cut-through proxy for faster processing of packets established sessions
   iv) Optional VPN support
   v) Stateful packet filtering

vi) NAT support
vii) Option for IPSec termination
viii) Option for failover/hot standby configuration
ix) Version 5.2 and higher supports use of SSH for encrypted remote management

c) The Internal Firewall will be a Cisco PIX 520 UR v 5.3(1) with six 10/100 Ethernet interfaces. It was chosen for the same capabilities as the external firewall in addition to those listed here:
   i) Ability to have six Ethernet interfaces
   ii) Option for URL filtering to screen access to Internet sites deemed inappropriate for business use (when used on conjunction with NetPartners WebSENSE)

d) The VPN device will be a Cisco VPN 5001. This device was selected because of the following capabilities which will provide for secure access from suppliers, partners, and remote users:
   i) Over 40 Mbps throughput
   ii) Up to 1500 simultaneous connections
   iii) Hardware-based d/encryption and key management
   iv) User and group level management
   v) 2 10/100 interfaces
   vi) Supports RADIUS authentication
   vii) Supports multiple client platforms including Windows, Linux, Solaris and Mac
   viii) DNS and WINS redirection
   ix) Tunneling protocol: IPsec
   x) Key management: IKE
   xi) Authentication options: IPsec ESP or AH using MD5 digital signature or SHA
   xii) Encryption options: IPsec ESP using DES or 3DES

6) Integrity assurance procedures and devices to provide additional layers of security in the form of alerts and activity logging.

Network access controls:

a) Ingress filtering: Deny traffic that's illogical (e.g. internal network addresses should never be seen coming from the outside.) Controlled access to devices that is restricted by source address could potentially be attained by outside entities if this measure is not taken.

b) Egress filtering: Reduces risk of misdirected transmissions of company data that is intended to stay within the company network.

c) Access control lists for dangerous protocols / services. Not only will restricting dangerous network service ports protect internal servers, but some services (e.g. telnet and ICMP) could be targeted directly against the defense devices themselves.

Administrative access restriction:

d) Require passwords for access (administrative or otherwise) to defense devices, and require that those passwords be stored encrypted on those devices.

e) Restrict access to security control and security monitoring devices by source address.

f) Encrypt remote access via SSH where possible; otherwise require local (console) access for device administration.

g) Restrict physical access to all defense devices behind locked doors!

Logging:

h) Log actual logons, and failed logon attempts to all perimeter devices.

i) Log suspicious activity. All perimeter devices (routers and firewalls) will log suspicious and denied access to a central syslog server located on the DMZ. The syslog server will review activity alerts from the firewalls and routers, and in turn send alerts to network administrators.

Intrusion detection:

j) IDS devices (ISS Real Secure) will be deployed on a number of the segmented networks, and will send alerts to their management server. All IDS network devices deployed will have two network interfaces. One interface will be connected to a switch on the target network, but will NOT have an IP address. It will merely be listening on that network, but will not be visible to other devices, and will not have the IP stack enabled on that interface. A second interface will connect directly to a switch/VLAN which connects to the IDS management station. The management station will also be dual-homed, but with IP enabled on both interfaces – one interface to receive alerts from IDS sensors, and the other to send alerts to network administrators, and to allow for remote management and log review.

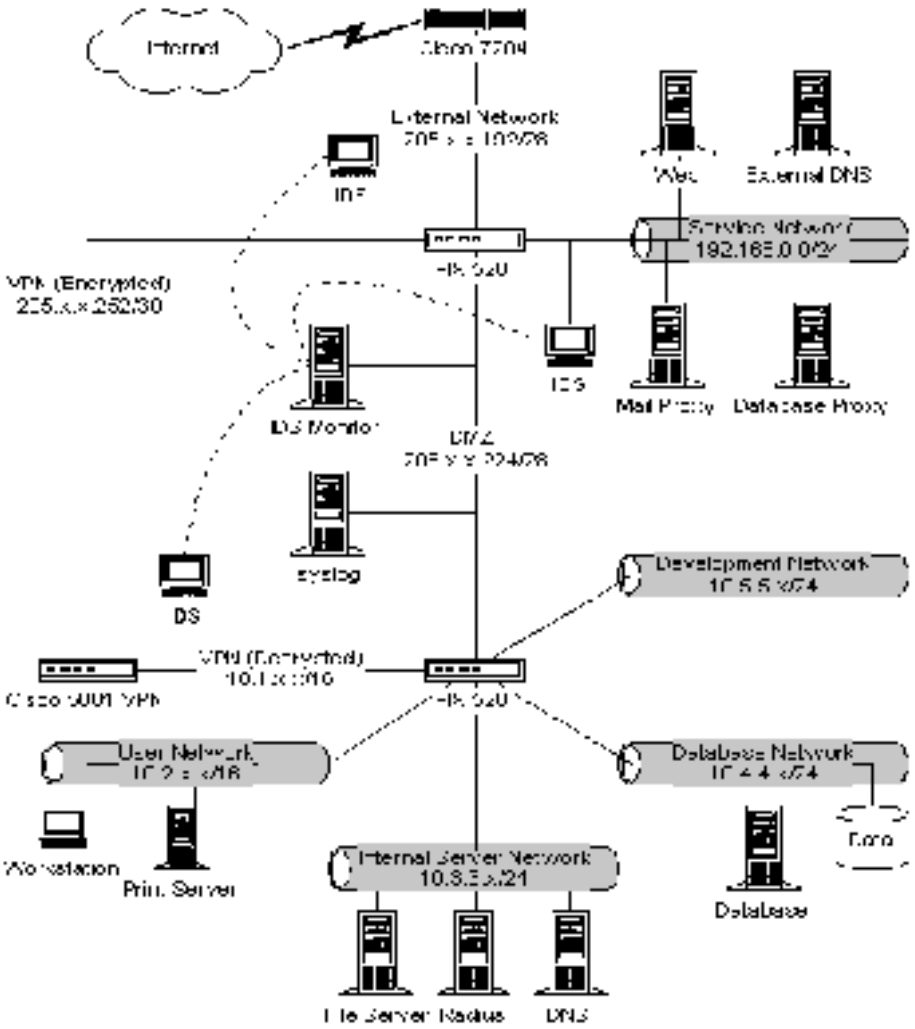


Figure 3 - Entire network layout including IDS and syslog. (Switches and VLANs omitted for readability.)

Assignment 2 - Security Policy (25 Points)
Based on the security architecture that you defined in Assignment 1, provide a security policy for AT LEAST the following three components:

- Border Router
- Primary Firewall
- VPN

You may also wish to include one or more internal firewalls used to implement defense in depth or to separate business functions.
By 'security policy' we mean the specific ACLs, firewall ruleset, IPSec policy, etc. (as appropriate) for the specific component used in your architecture. For each component, be sure to consider internal business operations, customers, suppliers and partners. Keep in mind you are an E-Business with customers, suppliers, and partners - you MAY NOT simply block everything!
(Special note VPNs: since IPSec VPNs are still a bit flaky when it comes to implementation, that component will be graded more loosely than the border router and primary firewall. However, be sure to define whether split-horizon is implemented, key exchange parameters, the choice of AH or ESP and why. PPP-based VPNs are also fully acceptable as long as they are well defined.)
For each security policy, write a tutorial on how to implement each ACL, rule, or policy measure on your specific component. Please use screen shots, network traffic traces, firewall log information, and/or URLs to find further information as appropriate. Be certain to include the following:

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

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

This assignment will be laid out in the following sequence:
1. Brief review and annotation of dangerous ports
2. Protocol matrix for each device
3. Guidelines for applying rules to each device to support the matrix
4. Commands for applying rules to each device
5. Methods for testing the rules to be applied to each device

---

In general, the approach to applying access control lists (ACLs) and rule sets to defend against unwanted traffic will be as follows: *permit only what is needed and deny everything else*. (Whether there is a known vulnerability from that port/service or not, this is the most secure approach.)

Because this approach results in not addressing many dangerous protocols specifically in the ACLs themselves, an analysis of the SANS top ten security recommendations is provided to support the goals of this exercise. Though most of the services/ports mentioned in the SANS list may not be specifically referenced in ACLs later in this assignment, their risks are addressed by the fact that most dangerous ports are not permitted, and those risky ones which need to be permitted are permitted in a controlled fashion.

Annotation of the SANS top ten (eleven)

The SANS top ten blocking recommendations in the first column is quoted from http://www.sans.org/topten.htm:

SANS Top Ten	Annotation and analysis
1. Block "spoofed" addresses-- packets coming from outside your company sourced from internal addresses or private (RFC1918 and network 127) addresses. Also block source routed packets.	• Blocking invalid address ranges from entering the network will reduce the risk of DoS attacks that cause internal computers to flood each other with traffic since replies are directed to the spoofed (i.e. internal) source address. DoS attacks typically take advantage of ICMP functions, which is why ICMP will be denied inbound as well. • Source routed packets contain extra information in the header which is intended to inform routers how to route the packets. This is useful in some networks, but crafty hackers can use this 'feature' to bypass routes as they've been configured to use round about access to a target. As a secondary precaution to source routed packets, the access lists will be secured as tightly as possible to avoid any such security holes.
2. Login services-- telnet (23/tcp), SSH (22/tcp), FTP (21/tcp), NetBIOS (139/tcp), rlogin et al (512/tcp through 514/tcp)	• Telnet allows users to access certain hosts to view/change configurations and file structures, and passes communications (including passwords) in clear text (unencrypted). Telnet can support file transfers to and from that target. Telnet will be allowed to the internal interface of the router for remote administration, but the source IP will be restricted. Telnet will be denied on the external interface. • FTP risks are similar to those of telnet in that communications are clear text, and FTP controls allow for exploration and modification of files structures and file transfers. FTP will be allowed outbound from the internal networks, but only valid replies to FTP requests will be allowed to return. • SSH is a preferred method of text-based remote access because it provides encryption, but SSH is not free from weaknesses. There's a known buffer-overflow vulnerability with certain versions of the service. Due to the risks, and the fact that Cisco routers do not currently support the use of SSH (PIX firewalls with IOS version 5.3 do) there will be no need to allow this protocol on the external networks except for the DMZ. • Rlogin, and rexec allow for remote execution of commands on the target host running the rexec daemon. A client transmits a message specifying the user name, the password, and the name of a command to execute, and with the right permissions, the command is run. The rexec daemon also transmits unencrypted passwords. • NetBIOS will be in use on the internal network to allow for connectivity and file sharing among Windows-based computers, but there will be no need for it on the external networks.
3. RPC and NFS-- Portmap/rpcbind (111/tcp and 111/udp), NFS (2049/tcp and 2049/udp), lockd (4045/tcp and 4045/udp)	• Remote Procedure Calls (RPC) allow for remote control and execution code which already exists on a target system or has been planted there by a hacker. On various operating systems including Linux, UNIX, and Windows, vulnerabilities range from directory access to complete control of a system. • NFS and lockd are used to control file access for network file systems. Vulnerabilities include creating or deleting files remotely.
4. NetBIOS in Windows NT -- 135 (tcp and udp), 137	• Windows NetBIOS provides access to shared network resources. Such resources should really only be shared by approved network servers which have been hardened to allow only appropriate access.

(udp), 138 (udp), 139 (tcp). Windows 2000 – earlier ports plus 445(tcp and udp)	Blocking all NetBIOS ports at network access points where there should be no such sharing will drastically reduce risk. Additional steps should be taken to eliminate shares and/or disable network shares for all systems not requiring them.
5. X Windows -- 6000/tcp through 6255/tcp	• X Windows is a protocol which supports remote control of a target machine. X Windows can be an attractive tool since it allows the remote user to see the target GUI screen. Vulnerabilities inherent to X Windows include the text being passed in the clear (passwords included) and the ability to monitor what is being typed at the keyboard on the remote machine. Thus if a hacker is monitoring a remote computer, it's possible for him/her to capture the password that's being typed in even if it does not appear on the screen.
6. Naming services-- DNS (53/udp) to all machines which are not DNS servers, DNS zone transfers (53/tcp) except from external secondaries, LDAP (389/tcp and 389/udp)	• Naming services are a critical function required to enable Internet users to find the company's web site. Name servers resolve host names to IP addresses. When a user explores the Internet for a host by name, his/her primary DNS server queries other DNS servers for the IP address of the target site. This is a necessary function of browsing the web with host names. Hackers can take advantage of this function by crafting packets that will send erroneous data to a name server along with a query. The name server adds that erroneous data to its database and shares it with anyone who requests it. Our perimeter device will block zone transfers. In addition the PIX has a built in function called DNS Guard to screen multiple replies to a query.
7. Mail—SMTP (25/tcp) to all machines, which are not external mail relays, POP (109/tcp and 110/tcp), IMAP (143/tcp)	• Part of the problem with these mail services is that commands are sent along with messages from one server to another telling the destination server what to do with the message. The possible control options outnumber what a company's server should do. Our PIX firewalls will use a "fixup" function to block unwanted controls. All perimter devices will restrict mail to just SMTP, and block POP and IMAP.
8. Web—HTTP (80/tcp) and SSL (443/tcp) except to external Web servers, may also want to block common high-order HTTP port choices (8000/tcp, 8080/tcp, 8888/tcp, etc.)	• HTTP is the common browsing protocol on the Internet, but many controls and scripts can be run across port 80 including java scripts and Active X controls. The default "fixup" function of the PIX will help eliminate some of the risks involved with allowing HTTP. The PIX firewall has options for filtering Active X, Java, and even URLs, but doing so can limit the functionality of users so they will not be used in this case.
9. "Small Services"-- ports below 20/tcp and 20/udp, time (37/tcp and 37/udp)	• By default, many host systems have these ports open though they may not be in use. Leaving these ports open is just leaving a back door for a hacker to exploit. Blocking these ports at network access points reduces some of the risk of not being able to harden all systems.
10. Miscellaneous-- TFTP (69/udp), finger (79/tcp), NNTP (119/tcp), NTP (123/tcp), LPD (515/tcp), syslog (514/udp), SNMP (161/tcp and 161/udp, 162/tcp and 162/udp), BGP (179/tcp),	• This group of protocols offers a range of tools to hackers, including probing for information transferring files. Finger for example can be used to probe systems running a finger daemon to find accounts for users who have never logged on… accounts that tend to have easy-to-guess passwords. TFTP can be used to transfer files to or from a compromised system. Listening to SNMP or BGP packets can give information about a network to allow an attacker to plan a more destructive attack. TFTP does not provide for user authentication, and is a major threat for stealing password files to be cracked off-line.

SOCKS (1080/tcp)	
11. ICMP—block incoming echo request (ping and Windows traceroute), block outgoing echo replies, time exceeded, and destination unreachable messages **except** "packet too big" messages (type 3, code 4). (This item assumes that you are willing to forego the legitimate uses of ICMP echo request in order to block some known malicious uses.)	• ICMP is a collection of tools for exploring networks and verifying that hosts and routes are working properly, and for transmitting error codes to assist with successful transmissions. Many vulnerabilities exist with this utility including network mapping, and several denial of service attacks.

Other protocols which will be in use in this network:

1. IPSec	• IPSec can have its own instability issues, though the protocol itself has not been known be a tool for security exploits or to deny service to networks. SecurityFocus.com does list one vulnerability related to mishandling of AH and ESP packets, but that's limited to running IPSec on OpenBSD.
2. SQLNet	• Many vulnerabilies with SQL tend to do with password weaknesses, and other security holes in the database or the operating system that hosts the database. The SQLNet port per se is not , and not something which perimeter devices can control. An old DoS vulnerability exists which targets port 1433 with a crafted packet, but our internal firewall will restrict access to port 1521.

General configuration approaches for all devices:

- To protect from spoofing, the router (and to some extent the firewall) will *deny all that is known to be invalid.*
- To protect from the use of unsafe protocols, the router and firewall will use the opposite approach, that being to *permit only that which we know to be desired.*

This approach towards protocol handling is more controlled and thus more secure, and alleviates the worry of missing something in our "deny" list, or opening ourselves to a yet-to-be discovered vulnerability of some obscure service. The other advantage to this approach is that the ACL will be shorter, requiring less processing cycles on the router and firewall. The list of the protocols we want to allow is much smaller than the list we would want to deny.

General traffic management tips:

- Services permitted inbound (from a lower security area to a higher one) are only permitted to a specific address or range of addresses.
- Return traffic is that which is replying to traffic which originated from the other direction.
- To simplify things in some cases, return traffic will be permitted inbound, to any destination.
- Outbound traffic is permitted from particular source addresses or ranges to any address.
- In the case of the router, most rules will be applied to the external, serial interface while others would only apply to the internal, Ethernet interface.

One of the first steps in detailing the security policies to be applied to the perimeter defense devices is to detail the protocols required for access to each network or host. For each device, a protocol matrix will be created to map which protocols will be allowed from where, to where. In some cases, one protocol matrix is sufficient to clarify access controls for a device. In other cases, each network connected to a device may need its own matrix.

Configuring Guidelines and Tips for the Cisco Router

Configuration tips:

- Disable all unneeded router services that can cause security loopholes.
- Some routing services need to be disabled on individual interfaces, not the entire router.
- Each interface can only have one access list applied inbound and one outbound, so and ingress filter which blocks source IP addresses will be the same one that blocks unwanted services (by only permitting what's desired).
- Access lists are scanned from the top down. The first match is the one that's used.
- Broader deny statements are put at the top to flush out the garbage, then specific permits are listed to allow only what is needed.
- An implicit "deny any any" is the last statement in an ACL. This statement denies anything that wasn't explicitly permitted in previous statements of that ACL.
- When a deny statement is used in an ACL, it should be followed by applicable permit statements, otherwise the implicit deny all will cause the ACL to effectively block everything.

Syntax tips for using subnets in ACLs:

- When defining a range of hosts for a subnet, instead of using the subnet mask as you would normally see in other contexts, it's written differently and is called a "wildcard". In a wildcard, all the bits are reversed from the subnet. Instead of the high-order bits being set (to one), the low-order bits are set. The following table shows an example:

Subnet mask for a /24 network				Corresponding wildcard for that network			
Octet 1	Octet 2	Octet 3	Octet 4	Octet 1	Octet 2	Octet 3	Octet 4
255	255	255	0	0	0	0	255
11111111	11111111	11111111	00000000	00000000	00000000	00000000	11111111

- In a /24 network the first 24 binary bits of the 32 bit netmask are all ones, and the rest of the bits are zeros. Notice that each binary bit in the subnet mask that was a 1 is switched to a 0 in the wildcard, and vice versa.

Subnet mask for a /28 network				Corresponding wildcard for that network			
Octet 1	Octet 2	Octet 3	Octet 4	Octet 1	Octet 2	Octet 3	Octet 4
255	255	255	240	0	0	0	15
11111111	11111111	11111111	11110000	00000000	00000000	00000000	00001111

- The second example is a bit more complex. In the fourth group of eight bits (octet) of the netmask only the first four bits are set. Thus in the fourth octet of the wildcard only the last four bits would be set.

Key words used in the access lists below:

- "any" does really mean any IP address.
- "host" means only one IP address.
- "range" means a range of IP addresses.
- "eq" means "equal to" the following port.
- "access-list" statements are used to add lines to an ACL.
- "access-group" statements are used to apply an ACL to an interface.

Preliminary steps:

- Log into the router and enter enabled mode by typing "enable".
- Enter the password for enable mode.
- Enter "config" mode by typing "config terminal" or "conf t" for short.

Services Permitted Through the Perimeter Router:

Service	Source	Destination	Port	Type	In/Out/Return
SMTP	Internet	Mail proxy NAT	25	Tcp	Both
DNS	Internet	External DNS NAT	53	Udp	In
	Internet	Internal DNS NAT	53	Udp	In
	Internet	Internal DNS NAT	53	Udp	Return
HTTP	Internet	Web server NAT	80	Tcp	In
	Internet	Web server NAT	443	Tcp	Both
	Internet	Any	80	Tcp	Return
SSL	Internet	Any	443	Tcp	Return
	Internet	Web server NAT	443	Tcp	In
FTP	Internet	Internal NAT range	20,21	Tcp	Return
IPSEC	Internet	Vpn		Tcp	Both

Apply the commands below to the router in config mode:.

```
hostname PerimeterRouter
```

Set passwords and enable password encryption.

```
service password-encryption
enable secret <your_password_here>
```

Enable logging.

```
logging 205.x.x.237
logging buffered 16384
logging trap debugging
```

Disable all unneeded services.

```
no service finger
no service pad
no service tcp-small-servers
no service udp-small-server
no snmp-server location
```

```
no snmp-server contact
no ip http server
no ip bootp server
no ip source-route
no ip name-server
no ip domain-lookup
```

Enable useful services.

Make the router CIDR compliant and enable the use of subnetting.

```
ip classless
ip subnet-zero
```

Ensure that telnet sessions left hanging by a remote system that crashed will be closed.

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

Set the default route to go out the serial interface to the Internet.

```
ip route 0.0.0.0 0.0.0.0 Serial 0/0
```

Apply the following commands to each interface by typing "int s0" or "int e0" prior to entering the list of commands.

Reduce risk of spoofing or session hijacking:

```
no ip redirects
```

Reduce the risk of DoS attacks:

```
no ip directed-broadcast
```

Reduce the risk of leaking internal MAC addresses to the outside world.

```
no ip proxy-arp
```

Reduce the rist of allowing details of this router, including routing tables, to be shared

```
no cdp enable
```

Apply this command to the external interface only to reduce the risk of network mapping.

```
no ip unreachables
```

Create the INBOUND access list for the outside interface

Ingress, anti-spoofing filter:

```
access-list 100 deny ip 0.0.0.0 0.255.255.255 any LOG
access-list 100 deny ip 10.0.0.0 0.255.255.255 any LOG
access-list 100 deny ip 127.0.0.0 0.255.255.255 any LOG
access-list 100 deny ip 172.16.0.0 0.15.255.255 any LOG
access-list 100 deny ip 192.168.0.0 0.0.255.255 any LOG
access-list 100 deny ip 205.x.0.128 0.0.0.128 any LOG
```

Protect against LAND attacks (200.x.x.1 is the address of the external, WAN interface):

```
access-list 100 deny tcp 200.x.x.1 0.0.0.0 200.x.x.1 0.0.0.0 LOG
```

Permit IPSec and IKE to the VPN device:

```
access-list 100 permit 50 any host 205.x.x.254
access-list 100 permit 51 any host 205.x.x.254
access-list 100 permit udp any eq 500 host 205.x.x.254 eq 500
```

Permit DNS queries to our public DNS server and queries from our private one.

```
access-list 100 permit udp any host 205.x.x.203 eq 53
access-list 100 permit udp any host 205.x.x.227 eq 53
```

Permit SMTP to our public mail server proxy.

```
access-list 100 permit tcp any host 205.x.x.202 eq 25
```

Permit WWW to our public web servers and replies to our internal networks:

NOTE: the "est" on the end denotes "established" connections, i.e. ones which originated from behind the router and are being replied to.

```
access-list 100 permit tcp any host 205.x.x.201 eq 80
access-list 100 permit tcp any range 205.x.x.231-235 eq 80 est
```

Permit SSL to the public web server and replies to our internal networks:

```
access-list 100 permit tcp any host 123.123.10.30 eq 443
access-list 100 permit tcp any range 205.x.x.231-235 eq 443 est
```

Although there is an implicit "deny any any" statement inserted by the router anyway, we want to log any access attempts that fall outside the scope of the statements above:

```
deny any any log
```

Apply this ACL to the external interface. 'S0' is our external interface.

```
int s0
ip access-group 100 in
```

Optionally, create the REFLEXIVE access list

The access list below presents some new concepts: Named ACLs, and reflexive ACLs. Reflexive access keeps track of connections which originate internally so as to cross-reference, or "reflect" on return traffic. In effect, this is a "dynamic ACL" which modifies it's table of source and destination addresses and ports each time a connection is initiated. This access is here to describe how it's possible to do this – it is not intended for use in this network because our firewalls will be keeping track of connection states. Using reflexive ACLs in this environment may be overkill and could cause performance degradation in times of high traffic.

```
ip access-list extended outbound
permit tcp range 205.x.x.231-235 any eq 80 reflect trafficout
permit tcp range 205.x.x.231-235 any eq 20 reflect trafficout
permit tcp range 205.x.x.231-235 any eq 21 reflect trafficout
```

Apply the above access list outbound on the external (s0) interface:

```
int s0
ip access-group outbound out
```

Tell the router to evaluate return traffic based on the dynamic "trafficout" ACL created by the outbound ACL:

```
ip access-list extended inbound
evaluate trafficout
```

Apply the reflexive access list to the external (s0) interface:

```
int s0
ip access-group inbound in
```

Create the access list for the internal interface for OUTBOUND traffic

Since this router will not support encrypted remote management, we want to limit telnet access to only come from the administrator's workstation.

As an extra security step, we'll explicitly deny echo replies from leaving the internal network. PING (echo) will be allows outbound since it's a very useful tool for determining of public hosts are active. However, a crafty cracker might try to use the echo reply protocol to scope out the internal network by explicitly denying it, we reduce that theat.

.

```
access-list 111 permit tcp 205.x.x.228 205.x.x.193 eq telnet
access-list 111 deny tcp any any eq telnet
access-list 111 deny icmp any any eq echo-reply
access-list 111 permit ip any any
```

The final statement might seem in conflict with the outbound access list applied to the external interface, but in fact they work in tandem. All traffic entering the internal interface will be screened again by the other ACL before it leaves the router. Incidentally, we do not want to log that "any any" statement since it would generate far to much information.

Apply this ACL to the internal interface:
Note: Though this is being referred to here as an "outbound" access list (because it's controlling traffic headed out of the network), it's applied to traffic coming "in" to the internal interface.

```
int e0
ip access-group 111 in
```

Secure the console and virtual terminal (telnet) ports

It is important to limit access to the VTY ports used for telnet access with an ACL. By default there are no access controls on any of the VTY ports. If left this way and a password is applied to the VTY port the router would be wide open to all comers to attempt a brute force crack against the password. A previously applied ACL will limit source IP addresses for telnet access. The lines below will set to the router to disconnect the console port, aux port, and VTY connections that have been idle for more than 5 minutes and 0 seconds. In addition, passwords will be set, and login will be required.

```
line console 0
exec-timeout 5 0
password <password>
login
line aux 0
exec-timeout 5 0
password <password>
login
line vty 0 4
exec-timeout 5 0
password <password>
login
```

Configuration Guidelines and Tips for the PIX Firewalls

PIX Firewall configuration tips (applicable to both the external and internal firewalls):

- By default, all ports are denied until explicitly permitted.
- At least one interface needs to have security 0, and one has to have security 100. The inside interface should be security 100, and the outside security 0. In short, these security levels tell the firewall which interface would be aware of which others, and which way traffic should be allowed to flow unless specified otherwise. Interfaces with higher security levels, such as the inside, will be aware of networks connected lower lever interfaces by default, but not the other way around. To allow devices on the outside interface for example be aware of any network found off the inside interface, a special "static" statement will be required.

Static statements:

- Static statements tell the PIX which hosts/subnets should be visible to which other networks, and how.
- The syntax of static statements with NAT enabled are different from those where NAT is not enabled in which network's address/range is where based on its relative security level. For example:
  1. When NAT is enabled: static (high_int, low_int) **low_ip** high_ip netmask mask
  2. When NAT is disabled: static (high_int, low_int) **high_ip** high_ip netmask mask
- The static statements in version 5.2 and later enable a new functionality of parameters following the static statements. The TCP Intercept Feature **protects internal hosts from**

TCP SYN floods by maintaining a table of SYN connections, and sending ACKs to the source from the PIX. If the source replies, then the PIX passes that traffic on to the internal host. The parameters allow the maximum connections as well as the size of the SYN table to be configured manually. If the maximum number of half-open SYN connections is reached, the PIX will not pass any more traffic to the server until those connections time out. Setting this number too low, can create a self-inflicted DoS, but it does protect the server from SYN floods, and services would resume when the flood stops. This feature will be implemented for all servers on the service network.

Conduit statements:

- Conduit statements are required to allow traffic to pass from one host or subnet to another. The syntax of conduit statements requires that the destination address is listed first, followed by the source address (note that this is the opposite from the ACLs for the router). Other syntax details for specifying protocol and port are noted below in the specific example.
- When used with a static command statement, a conduit command statement permits users on a lower security interface to access a higher security interface. When not used with a static command statement, a conduit command statement permits both inbound and outbound access.
- If you use PAT (Port Address Translation), you cannot use a conduit command statement using the PAT address to either permit or deny access to ports.
- A DNS server on a higher level security interface needing to get updates from a root name server on the outside interface cannot use PAT (Port Address Translation). Instead, a static command statement must be added to map the DNS server to a global address on the outside interface.
- Conduit commands are processed in the order they're entered in the configuration. Removing a single conduit command can have undesirable effects on the traffic control. When removing or re-sequencing conduits, follow these steps:
  1. Copy the existing commands from the terminal session to a text editor like notepad.
  2. Edit the command list.
  3. Clear the entire existing conduit list from the PIX with the "clear conduit" command.
  4. Verify that the list is gone with "show conduit".
  5. Copy and paste the entire new list to the configuration.
  6. Verify that the list has been entered correctly with "show conduit".
  7. Verify that the changes you made are having the desired effect.

Configuring the External Firewall:

Protocol matrix showing services permitted through the external firewall (allowing return traffic is implied, and will be tracked by the stateful connection table)

Service	*Source*	*Destination*	*Port*	*Type*	*In/Out*
SMTP	Internet	Mail proxy	25	Tcp	Both
DNS	Internet	External DNS	53	Udp	In
HTTP	Internet	Web server	80	Tcp	In
SSL	Internet	Web server	443	Tcp	In
ESP	Internet	VPN	50	-	Both
AH	Internet	VPN	51	-	Both
ISAKMP	Internet	VPN	500	Udp	Both
SMTP	DB Proxy	Internal Database NAT	25	Tcp	Both
DNS	Mail Proxy	Internal Mail NAT	53	Udp	In
HTTP	DMZ	Any	80	Tcp	In
SSL	DMZ	Any	443	Tcp	In
DNS	Internal DNS NAT	Internet	53	Udp	In

Configuring the PIX firewall to adhere to the protocol matrix above:

Name the interfaces and define security levels.

```
nameif e0 outside sec0
nameif e3 sevice sec40
nameif e2 vpnext sec60
nameif e1 inside sec100
```

Establish a password for configuration level access (encrypted by default on the PIX)

```
enable password mypa55word encrypted
```

Set the password for telnet access.

```
passwd mytelnetpa55word encrypted
```

Set a hostname.

```
hostname ExternalPIX
```

The following fixup statements are default on this version of PIX software.

These default ports conveniently cover all the non-encrypted protocols used in this network. This command features runs an "Adaptive Security Algorithm" for each port listed based on different port numbers other than the defaults to ensure that only appropriate communications are taking place. Here are some specific examples of how what impact this feature can have:

- The "strict" option for the ftp fixup prevents browsers from sending embedded commands to a server. This control would also restrict clients to sending only PORT commands, and servers to sending only the 227 command. The PIX also verifies that these commands are not part of an error string. Since there are no FTP servers inside our network, this won't be necessary.
- The smtp fixup enables the Mail Guard feature, which only lets mail servers receive approved commands of HELO, MAIL, RCPT, DATA, RSET, NOOP, and QUIT. All other commands are rejected with the "500 command unrecognized" reply code.

```
fixup protocol ftp 21
fixup protocol http 80
fixup protocol h323 1720
fixup protocol rsh 514
fixup protocol smtp 25
fixup protocol sqlnet 1521
fixup protocol sip 5060
```

Establish some settings for ease of use and editing. The 'names' command (enabled by default) allows for the use of nick-names for ip addresses for readability.

TIP: Names it also allows you to change the address of a device in the configuration with one statement instead of having to locate each line that referenced that address. TIP: The list of names can be cut and pasted into configuration for other PIXes on the network.

```
names
name 205.x.x.x dns_ext_nat
name 205.x.x.x mail_ext_nat
etc…
```

Enable logging to the syslog server.

The following commands turn logging on and establish the address of the logging host. They also set the level messages to be held in the buffer (errors or higher) and those to be sent to the syslog (warnings or higher). Logging to the console would greatly impact the performance of the PIX, and is disabled by default.

```
logging on
logging host inside 205.x.0.237
logging facility 20
logging trap warnings
logging buffered errors
no logging console
```

Set speed and mtu on all interfaces. Each interface will be connected directly to a 100Mb switch or directly to another NIC capable of 100Mb full duplex. A maximum transmition unit size of 1500 bytes defines an ethernet network.

```
interface outside 100full
interface service 100full
```

```
interface vpnext 100full
interface inside 100full
mtu outside 1500
mtu service 1500
mtu vpnext 1500
mtu inside 1500
```

Set the IP address and subnet maks for each interface.

```
ip address outside 205.x.0.194 255.255.255.128
ip address service 192.168.0.194 255.255.255.0
ip address vpnext 205.x.0.253 255.255.255.252
ip address inside 205.x.0.225 255.255.255.248
```

CONFIGURE NAT

Enable NAT for the service network and no other.

```
nat (service) 1 0.0.0.0 0.0.0.0 0
nat (inside) 0 0.0.0.0 0.0.0.0 0
nat (vpnext) 0 0.0.0.0 0.0.0.0 0
```

CONFIGURE STATIC STATEMENTS

The syntax for a network with no NAT is: static (Higher_Secuity_Interface, Lower_Security_Interface) High_network High_network netmask High_network_mask

TIP: If you omit the netmask portion of the statement, the PIX will assume the netmask to be 255.255.255.255 (a single host).
The next lines let the service network be aware that the network range of the DMZ exists and where. Without this statement, traffic from the service network to the DMZ would be dropped.

```
static (inside,service) 205.x.x.224 205.x.x.224 netmask
255.255.255.240
static (inside,outside) 205.x.x.224 205.x.x.224 netmask
255.255.255.240
static (vpnext,outside) 205.x.x.252 205.x.x.252 netmask
255.255.255.252
```

Define static NAT for specific servers/clusters on the service network. Use static statements to map NAT addresses between public and private. The systax for a netowrk with NAT enabled is a bit different: static (Higher_Secuity_Interface, Lower_Security_Interface) High_network_NAT Low_network_host_address netmask High_network_mask. Since these are all host addresses, the netmask portion of the statement can be left off.

```
static (service,outside) 205.x.0.200 192.168.0.200 10000 1000
static (service,outside) 205.x.0.201 192.168.0.201 10000 1000
static (service,outside) 205.x.0.202 192.168.0.202 10000 1000
```

The previous lines also implement the TCP Implement Feature (mentioned earlier) to set the maximum number of connections for each server, and the maximum number of TCP SYN connections held for verification before being passed to the server.

CONFIGURE CONDUITS

Use conduit statements to permit specific ports to the public NAT address. After changing or removing a conduit, use the "clear xlate" command to clear the translation table. Conduit statements use the following syntax: conduit <permit/deny> <protocol> <destination address [netmask]> <port> <source address [netmask]>
Note that the addresses used here are the actual addresses of the hosts on the service network, not their NAT addresses as they would appear on the outside interface. The PIX will know where they are and how to translate addresses appropriately. This allows secure access through the NAT address from the outside and access to the 192.x.x.x addresses when communicating from the inside.

For the web server...

```
conduit permit tcp host 192.168.0.200 eq www any
conduit permit tcp host 192.168.0.200 eq 443 any
```

For the DNS server...

```
conduit permit udp host 192.168.0.201 eq dns any
```

For the mail server...

```
conduit permit tcp host 192.168.0.202 eq smtp any
```

Define other conduits that don't relate to NAT. Allow perimeter router to send to the syslog server though the PIX.

```
conduit permit udp host 205.x.0.237 514 eq syslog host
205.x.0.193
```

Permit IPSec to pass through the PIX.

Note that in the matrix at the beginning of this section, it was specified that AH, ESP and ISAKMP all needed to be permitted. This single statement covers all those needs.

```
sysopt connection permit-ipsec
```

Allow connecting to the PIX using ssh… from the administrator's workstation only through a designated interface. This could be an address of an actual host on the DMZ or it could be a static NAT from the interior networks. Note: A hardware upgrade to support DES and 3DES will be necessary to support this feature. Also note that 5 minutes is the default timeout for an idle connection:

```
ssh 205.x.x.x 255.255.255.255 inside
ssh timeout 5
```

Another conduit example to use when testing connectivity with ICMP:

The following two commands will permit any host on the service network to initiate a ping to any host across any interface of the firewall. Without the second line, however, the reply would not be allowed back from a lower level interface such as the outside. To enable echo –replies from the outside to a particular host on the service network, the second statement is required.

```
conduit permit icmp any 192.168.0.0 255.255.255.0
conduit permit icmp host 205.0.0.201 any echo-reply
```

NOTE: When ICMP connectivity testing has completed, the ICMP ports should be closed again by removing these conduit statements with the "no conduit…" command.

SET ROUTES

Set the default route to the perimeter router off the external interface of the firewall.

```
route outside 0.0.0.0 0.0.0.0 205.x.x.193 1
```

NOTE: The PIX automatically builds its own routing table to the directly connected networks off each interface. Though it knows the relative location of all those network that does not mean it will pass traffic. Conduit statements are required to allow traffic to pass.

CONTROL ICMP TRAFFIC

Make the firewall less noticeable on the external interface by denying most icmp traffic without disabling ipsec. Cisco recommends that you grant permission for ICMP unreachable message type (type 3). Denying ICMP unreachable messages disables ICMP Path MTU discovery, which can halt IPSec and PPTP traffic. See RFC 1195 and RFC 1435 for details about Path MTU Discovery. icmp <permit/deny> <source address> [source mask] [type] <interface_name>

```
icmp permit 205.x.x.254 unreachable outside
icmp deny any echo-reply outside
```

Configuring the Internal Firewall:

To demonstrate a comparison of implementation methods, the Internal firewall will use access-lists instead of Conduit statements. NAT, Global, and Static settings will still need to be defined.

Access lists:

- Use of access-list and access-group commands **overrides conduit commands**.
- Quoting from Cisco's Command Reference for PIX 5.2, "*Cisco recommends that you do not use the access-list command with the conduit and outbound commands. While using these commands together will work, the way in which these commands operate may cause debugging issues because the conduit and outbound commands operate from one interface to another whereas the access-list command used with the access-group command applies only to a single interface. If these commands must be used together, PIX Firewall evaluates the access-list command before checking the conduit and outbound commands.*"
- Access lists are applied to traffic going **into an interface only**, unlike routers which can have one access-group applied inbound, and one outbound. So, on the matrixes below, they only need to define traffic coming in.

Access permitted from the "outside" Internet, DMZ, and service networks:

Service	***Source***	***Dest.***	***Port***	***Type***
SMTP	Mail server	Mail proxy	25	Tcp
SQLNet	DB Proxy	Database server	1521	Tcp

Access permitted from the internal VPN network.

Service	***Source***	***Dest.***	***Port***	***Type***
SMTP	10.1.x	Internal Mail	25	Tcp
DNS	10.1.x	Internal DNS	53	Udp
HTTP	10.1.x	Any	80	Tcp
SSL	10.1.x	Any	443	Tcp
SQLNet	10.1.x	Database	1521	Tcp
NetBIOS-ns,-dgm	10.1.x	10.x.x	137, 138	Tcp

Access permitted from the user network

Service	***Source***	***Dest.***	***Port***	***Type***
SMTP	10.2.x.x	Internal Mail Server	25	Tcp
DNS	10.2.x.x	Internal DNS Server	53	Udp
HTTP	10.2.x.x	Any	80	Tcp
SSL	10.2.x.x	Any	443	Tcp
SQLNet	10.2.x.x	Database	1521	Tcp
NetBIOS-ns,-dgm	10.2.x.x	10.x.x	137, 138	Tcp
DHCP client	10.2.x.x	DHCP server	68	Tcp
DHCP server	DHCP server	10.2.x.x	67	Tcp

Access permitted from the file server network

Service	***Source***	***Dest.***	***Port***	***Type***
SMTP	Internal Mail Server	Any	25	Tcp
DNS	Internal DNS Server	Any	53	Udp
HTTP	10.3.3.0/24	Any	80	Tcp
SSL	10.3.3.0/24	Any	443	Tcp
SQLNet	10.3.3.0/24	Database	1521	Tcp
NetBIOS-ns,-dgm	10.3.3.0/24	10.0.0.0/8	137, 138	Tcp
DHCP server	10.3.3.50	10.2.0.0/16	67	Tcp
DHCP server	10.3.3.50	10.5.5.0/24	67	Tcp

Access permitted from the database network

Service	***Source***	***Dest.***	***Port***	***Type***
SQLNet	10.x.x.x	Database	1521	Tcp
SQLNet	192.168.0.	Database	1521	Tcp

Access permitted from the "inside" development network

Service	*Source*	*Dest.*	*Port*	*Type*
IP	10.5.5.x	Any		

Many of the general configuration commands will be left out of this section because they're the same as for the external PIX. The important distinction here is that Access lists are used instead of conduit commands.

```
nameif e0 outside sec0
nameif e1 vpnint sec20
nameif e2 users sec40
nameif e3 fileservers sec60
nameif e4 database sec80
nameif e1 inside sec100
hostname InternalPIX
```

The access-lists below are easier to create and read if they use names…

```
names
name 192.168.0.201 svc_web
name 192.168.0.202 svc_mail
name 192.168.0.203 svc_dns
name 192.168.0.204 svc_dbproxy
name 10.3.3.201 internal_web
name 10.3.3.202 internal_mail
name 10.3.3.203 internal_dns
name 10.4.4.204 internal_db

ip addr outside 205.0.0.230 255.255.255.240
ip addr vpnint 10.1.1.1 255.255.255.0
ip addr users 10.2.2.2 255.255.255.0
ip addr fileservers 10.3.3.3 255.255.255.0
ip addr database 10.4.4.4 255.255.255.0
ip addr inside 10.5.5.5 255.255.255.0
```

The first global statement created the initial pool of addresses, the second designates the Port Address Translation (PAT) address for overflow from the global pool.

```
global (outside) 1 205.0.0.231-205.0.0.234
global (outside) 1 205.0.0.235
```

NAT is enabled for each internal interface for access to the outside.

```
nat (inside) 1 0 0
nat (database) 1 0 0
nat (vpnint) 1 0 0
nat (users) 1 0 0
nat (fileservers) 1 0 0
```

Here are just a couple example static statements from one internal network to another (not using NAT to reach each other).

```
static (fileservers,users) 10.3.3.0 10.3.3.0 netmask
255.255.255.0
static (inside,users) 10.5.5.0 10.5.5.0 netmask 255.255.255.0
static (inside,fileservers) 10.5.5.0 10.5.5.0 netmask
255.255.255.0
```

And here's an example of the static mappings from an internal host to an external statically assigned NAT address.

```
static (fileservers,outside) 205.0.0.232 internal_mail netmask
255.255.255.255
static (fileservers,outside) 205.0.0.233 internal_dns netmask
255.255.255.255
```

And of course, the default route…

```
route outside 0.0.0.0 0.0.0.0 205.0.0.225 1
```

Configuring the access-lists

Configure access from the outside network:

```
access-list outside permit tcp svc_mail internal_mail eq smtp
access-list outside permit tcp svc_dbproxy internal_db eq sqlnet
```

Apply that access-list to the interface

```
access-group outside in interface outside
```

Configure access from the internal vpn network:

```
access-list vpn permit tcp 10.1.1.0 255.255.255.0 internal_mail
eq smtp
```

NOTE: Remote users connecting through the VPN will be assigned addresses in the 10.1.x.x range. Hosts connecting form partner VPN sites will have addresses in the 10.6.x.x range. The partner hosts will not be permitted to connect with our internal mail server.

```
access-list vpn permit tcp 10.1.1.0 255.255.255.0 internal_mail
eq smtp
access-list vpn permit tcp 10.1.1.0 255.255.255.0 internal_dns eq
dnsix
```

NOTE: the following lines will permit remote access users to get to the designated web server on the internal network, but not to connect to any other web server that might exist on the 10.x network. The third line permits web access to the internet.

```
access-list vpn permit tcp 10.1.1.0 255.255.255.0 host
internal_web eq www
access-list vpn deny tcp any 10.0.0.0 255.0.0.0 eq www
access-list vpn permit tcp 10.1.1.0 255.255.255.0 any eq www
access-list vpn permit tcp 10.1.1.0 255.255.255.0 host
internal_web eq 443
access-list vpn deny tcp any 10.0.0.0 255.0.0.0 eq 443
access-list vpn permit tcp 10.1.1.0 255.255.255.0 any eq 443
```

Permit access to the database

```
access-list vpn permit tcp 10.1.1.0 255.255.255.0 database eq
sqlnet
```

Permit access to other internal hosts (note that split-horizon will be disabled, so remote users won't be able to find other remote users, or partner's hosts.

```
access-list vpn permit udp 10.1.1.0 255.255.255.0 10.0.0.0
255.0.0.0 eq netbios-ns
access-list vpn permit udp 10.1.1.0 255.255.255.0 10.0.0.0
255.0.0.0 eq netbios-dgm
access-list vpn permit tcp 10.1.1.0 255.255.255.0 10.0.0.0
255.0.0.0 eq 139
access-list vpn permit tcp 10.1.1.0 255.255.255.0 10.0.0.0
255.0.0.0 eq 445
```

Anyway, you get the idea….

One more example to show the contrast between the restrictions on the VPN network to those on the development network which will allow anything out of that network:

```
access-list inside permit ip 10.5.5.0 255.255.255.0 any
```

Apply that access-list:

```
access-group inside in interface inside
```

Configuring the VPN:

Configuration steps:

Configure the interfaces:
Ethernet interface 1 only passes IPSec traffic, so that needs to be the external interface. Ethernet 0 will be connected to the internal network.

Configuring Ethernet 0:

```
config ip ethernet 0
ipaddress=192.168.233.1
subnetmask=255.255.255.0
ipbroadcast=192.168.233.255
```

Configuring Ethernet 1:

```
config ip ethernet 1
ipaddress=206.45.55.1
subnetmask=255.255.255.0
ipbroadcast=206.45.55.255
```

Configure default IP parameters which apply to all interfaces:

```
Config ip default
Mode=routed
SplitHorizon=off
OutFilters
InFilters
exit
```

Define the default route for the internal connection:
Syntax for the route statements is as follows: <destination> <mask> <gateway> <metric> where the metric is the number of hops away for the gateway. The append command triggers appending to the (now blank) routing table.

```
edit config ip static
append 1
0.0.0.0 0.0.0.0 10.1.1.1 1
exit
```

Define the default route (IPSec Gateway) for the external connection.
The IPSec Gateway controls where the concentrator sends all the IPSec traffic out of the Ethernet 1 interface. In our case, this is the VPNEXT interface of the external PIX. To set the gateway, enter the "config general" command to configure these general system parameters:

```
config general
ipsecgateway=205.x.x.253
exit
```

IKE Policy.

Internet Key Exchange (IKE) parameters control how authentication takes place when a tunnel connection is first established.

The tunnel partners negotiate key exchange for the session within the parameters set on each one, looking for the most secure match possible.

IKE negotiation takes place in two phases.

- Phase 1 negotiation parameters are used to establish a secure, authenticated channel for the rest of the session.
- Phase 2 establishes the Security Associations for the session. Security Associations are policy rules that map to particular peers. These rules are maintained in a Security Association Database (SADB).

Phase 1 parameters are set globally in the IKE Policy section
Phase 2 parameters are set with the "Transform" parameter when VPN groups and partners are configured.

The protection command can be entered multiple times, in which case it would allow this device to negotiate a greater range of parameters with partners. The syntax for IKE Policy is as follows:
Protection = [MD5_DES_G1 | MD5_DES_G2 | SHA_DES_G1 | SHA_DES_G2]

There are three configurable parts of the protection separated by the underscore. The first part is the authentication algorithm to be used in Phase 1 of negotiation. MD5 stands for message-digest 5, and SHA is Secure Hash Algorithm – SHA is generally considered to be more secure.

The second part is the encryption algorithm, which can be DES or 3DES (triple DES) – 3DES being more secure since it runs the same DES algorithm three times.

The third part of the protection is the Diffie-Hellman group to be used for key exchange. Since larger numbers are used for Group2, it is considered the more secure of the two.

For the purposes of this network, these global settings will be set as high as possible, with an alternate protection level for clients who can't use 3DES.

```
config IKE Policy
Protection = SHA_3DES_G2
Protection = SHA_DES_G2
exit
```

The Transform parameter will be set below for each partner and group. Transform specifies the protection types and algorithms to be used for IKE client sessions (that which follows the authentication phase). The options are as follows:
Transform = [ESP(SHA,DES) | ESP(SHA,3DES) | ESP(MD5,DES) | ESP(MD5,3DES) | ESP(MD5) | ESP(SHA) | AH(MD5) | AH(SHA) |AH(MD5)+ESP(DES) | AH(MD5)+ESP(3DES) | AH(SHA)+ESP(DES) | AH(SHA)+ESP(3DES)]

ESP(MD5,DES) is the default setting

For this network we'll use the most secure for the partner connection, and less secure for remote users. Remote users may be connecting via modem and would see an extra performance hit if more processing power is needed to deal with more complex encryption algorithms.

For partner sites, we'll choose: AH(SHA)+ESP(3DES)

For remote users we'll choose: AH(SHA,DES)

Configure Site-to-Site connectivity:

Tunnel partners
The parameters to be defined here would first be negotiated with the administrator of the remote VPN connection at the partner site. Authentication and encapsulation methods need to be agreed upon, and the shared key needs to be securely shared and entered into each system. The address of the partners VPN termination point is entered here as well. The administrator of the partner's system would enter the address of the external interface of our VPN (205.x.x.254). The peer addresses define the range of that our VPN will expect to see come across the wire.

```
config Tunnel Partner VPN 1
transform= AH(SHA)+ESP(3DES)
sharedkey=thisismykey
partner=105.x.x.100
mode=main
peer=10.6.6.0/24  [this is an address range to be expected from the partner site]
localaccess=10.1.0.0/16
bindto=Ethernet 1
exit
```

IP parameters

```
config ip vpn 1
mode=routed
numbered=false
exit
```

Configure Remote User connectivity

VPN Groups
There are two options when defining IP address ranges for each group.

LocalIPNet assigns IP addresses to remote hosts from a subnet that must be unused elsewhere on the LAN. In our case, we know that there are not other hosts on the network in the 10.1.1.x range besides the inside of the VPN and one of the interfaces of the internal PIX.

The command "IPNet" determines which host ranges on the local network can be tunneled through to the partner sites. The IPNet statement can appear up to 64 times here.

The second choice would be to use the StartIPAddress command which tells the VPN to issue ip addresses starting at a certain number and working up.

The two examples below would have similar results:

```
config VPN group GIAC-Employee
startipaddress=10.1.1.2
```

or

```
localipnet=10.1.1.2/24
```

then

```
maxconnections=250
Transform= AH(SHA,DES)
ipnet=10.4.4.0/24
ipnet=10.3.3.0/24
dnsprimaryserver=10.3.3.33
exit
```

To add an additional layer of password security, the Cisco VPN 5000 series offers authentication with a RADIUS server or with SecurID. SecureID is a Security Dynamic proprietary system that requires the user to carry a small device to randomly generate a new unique digital key every

minute. The remote user could be required to enter their user ID, password, and SecurID code for access to the network. On the network side, the system requires ACE/Server software and SecureID tokens to perform the dynamic authentication.

SecureID can be required on either the user or group level by adding the following line:

```
SecureIDRequired=On
```

VPN Users

If so desired, this step would allow a group of users to use the same shared key for remote access. The format for the entry is: <Name> Config="group-name" SharedKey="<SharedKey>"

```
edit config VPN users
append 1
BubbaSmith Config="GIAC-Employee" SharedKey="Key4Bubba"
exit
```

Adding additional users can be easy with cutting and pasting from a text file to the console.

Enable logging:

```
Configure logging
Level=7
Enabled=On
LogToAuxPort=Off
LogToSysLog=On
SyslogIPAddress=205.x.x.237
SyslogFacility=Local5
```

Save the configuration to the system by typing "save".

Determine authentication and encryption methods to be used:

DEFINE THE ACCESS:

Group Permissions on VPN:

Permissions which apply to both Business Partners (10.6.6.x) and Remote Users (10.1.x.x):

Service	Source	Destination	Port	Type	In/Out/Return
SMTP	VPN	Mail server	25	Tcp	Both
DNS	VPN	Internal DNS	53	Udp	In
	VPN	Internal DNS	53	Udp	Return
HTTP	VPN	Intranet Web	80	Tcp	In
	VPN	Any	80	Tcp	Return
SSL	VPN	Intranet Web	443	Tcp	Both
	VPN	Any	443	Tcp	Return
SQLNet	VPN	Database server	1521	Tcp	Both
NetBIOS	VPN	One Server on Fsnet	137-8, and 445	Udp	Both
NetBIOS	VPN	One Server on Fsnet	139, and 445	Tcp	Both

Other permissions only to be granted to Remote Users:

Service	Source	Destination	Port	Type	In/Out/Return
NetBIOS	Remote users				
NetBIOS	VPN	One Server on Fsnet	137-8, and 445	Udp	Both
NetBIOS	VPN	One Server on Fsnet	139, and 445	Tcp	Both
FTP	VPN	Intranet FTP	20,21	Tcp	Return

Creating VPN access lists:

The VPN 5001 is another Cisco device which can accept access lists similar to the router and firewall in this network. The syntax here is slightly different from both the other devices. The router uses wildcards, the PIX uses subnet mask, and the VPN 5000 series uses CIDR notation (also called "slash" notation). With CIDR, the number of bits used for the network masks is entered after a forward slash. If the bits are left out, its assumed to be a single host. The syntax rules are as follows: Permit | deny <source_IP>/bits <destination_IP>/bits <protocol>

As with the ACLs on other devices, anything not explicitly allowed will be dropped.

With the VPN 5000 series, the access-list would be analogous to an "ip filter", and an access-group would be analogous to an "IPFilterOut" or "IPFilterIn" statement. Unlike routers which only allow one inbound and one outbound access-group to be applied to each interface, and the PIX which only allows on access-list inbound on each interface, the VPN allows up to four lists to be applied in each direction. This helps support the fact that there may be multiple VPN groups configured with multiple addressing schemes.

For simplification purposes, we'll rely on our firewall to block protocols on the internal network, and simply use the access lists here to apply basic ingress and egress filters to only allow what would be legitimate traffic and implicitly denying everything else.

Remember also that split-horizon was disabled in the "config ip default" section above.

Create an access list to be applied to traffic coming out of the interface to the LAN:

```
edit config ip filter "filter_out"
permit 10.1.1.0/24 0.0.0.0 ip
permit 10.6.6.0/24 0.0.0.0 ip
```

Create an access list to be applied to traffic coming into the interface from the LAN:
This list will also restrict telnet traffic for configuring the device.

```
edit config ip filter "filter_in"
permit 10.3.3.99 10.1.1.{99,100,101} tcp dst = telnet
deny 0.0.0.0 0.0.0.0 tcp dst = telnet
permit 10.3.3.0/24
permit 10.4.4.0/24 0.0.0.0 ip
```

Apply the ip filters to the internal interface:

```
config ip ethernet 0
Outfilters = filter_out
Infilters = filter_in
```

Testing filters and access control lists:

Establish connectivity.
Ping tests are a good way to verify routing, and also to verify that the "static" statements on the firewalls are configured properly. On some devices, ICMP may be blocked, so it could be useful to allow it for this initial test phase. Allowing ping will only verify that the path is valid, it won't test to see if other application ports are secure or even available. Using the network diagram, verify that each device is reachable from where it's supposed to be. For example, the servers on the service network should be reachable from the internal networks (and the echo-reply should come back). But servers on the outside network should not be able to ping into NAT addresses for the internal network.

If there's a question about what traffic is passing through a perimeter device, turn on debug on that device. With "debug icmp trace" on the PIX for example, you would see the echo including

source and destination, and the echo reply (if any) with the addresses as well. For viewing the passage of other traffic, there are other debug functions available like "debug packet <interface>". Use caution when using debug because it can affect traffic greatly since your device is using CPU time to show you what's passing through. Also be aware that you could see delays in transmission speeds with debug enabled.

Make sure it's functional!

Security is very important, but if people and devices can't function, the configuration is wrong. Certain devices should be tested specifically, like mail, web, dns, and database servers, but the real test of a network's functionality will come when the users get to it.

Verify the connections are secure.

CyberCop is one of many scanning tools available on the market. CyberCop is a full-featured product with a GUI interface by Network Associates with options for scanning, and mapping networks, as well as probing for specific vulnerabilities or categories of them. Scan reports and graphs can also be automatically generated.

In brief, here is how this product would be used to verify the ACLs applied to the perimeter router.

1. A workstation with CyberCop Scanner should be put on the outside of the router.
2. Another workstation with CyberCop Sentry should be put on the inside of the router.
3. Modify the IP addresses on the Scanner PC and the Sentry PC to match actual source and destination addresses referenced in the ACL. For example, the Scanner PC could take the address of the ISP (if the correct hardware is available to support that connection), and the Sentry could assume the address of a device on the external network, like the external address of the external firewall, or one of the NAT of servers on the service network.
4. When the addresses are set, initiate a scan from the Scanner to the Sentry through the device to see what ports were open. Compare the scan results to the protocol matrix for each device.

Verifying the ACLs and filters on the firewalls will follow the same methods, but will be more involved because tests need to be done from each interface to each of the other interfaces on that device.

Verifying the security *and functionality* of the entire network design should involve running scans across the entire network, not just across each individual device. These scan results should show that desired services are open where they are expected to be and that successful traffic transfer is occurring as well as showing that security holes are not open. Of course the ultimate test of functionality needs to be conducted by the actual users of the network.

Ideally some testing of protections against common attacks such as DoS attacks should be conducted in a controlled way because these attacks could cause devices and hosts to freeze or crash.

Assignment 3 - Audit Your Security Architecture (25 Points)
You have been assigned to provide technical support for a comprehensive information systems audit for GIAC Enterprises. You are required to audit the Border Router and Primary Firewall described in Assignments 1 and 2. Your assignment is to:

1. Plan the assessment. Describe the technical approach you recommend to assess your perimeter. Be certain to include considerations such as what shift or day you would do the assessment. Estimate costs and level of effort. Identify risks and considerations.
2. Implement the assessment. Validate that the Border Router and Primary Firewall are actually implementing the security policy. Be certain to state exactly how you do this, including the tools and commands used. Include screen shots in your report if possible.
3. Conduct a perimeter analysis. Based on your assessment (and referring to data from your assessment), analyze the perimeter defense and make recommendations for improvements or alternate architectures. Diagrams are strongly recommended for this part of the assignment.

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

Overview:

The assessment plan for the GIAC Enterprises network design will include the following phases of scanning, and evaluation, and recommendations:

The first phase of the assessment will include a port scan through each interface of the router and firewall to the other networks. Once hosts have been identified, the host list will be reviewed with network administrators from GIAC Enterprises To make final determinations of which host will be scanned further and/or targeted for DoS attacks or other probing.

The second phase will consist of testing hosts and devices fur susceptibility to known vulnerabilities.

GIAC Enterprises will be presented with a summary of scan results, and a detailed list of recommendations to make the network more secure.

Scheduling and pricing:

Though it is not expected that any of the devices/hosts being evaluated here will be rendered out of service by any of the tests, there's always a possibility. It's also possible for there to be an impact on network performance during some tests. Given those risk, this test will be schedule for off-peak hours for GIAC Enterprises.

The assessment will begin Saturday morning at 8am because most employees are not at work, and most customers of GIAC are restaurateurs, and would not be on line during that time.

A flat fee of $5000 will be charged for all aspects of this assessment including the delivered documentation. It is expected that the scanning can be completed in one day with the cooperation of the company's network administrators. Documentation will be delivered by the following Friday.

Scope of assessment:

The testing team will have the following permissions and limitations on intrusiveness during the testing phase:

- Permitted to attempt to gain access to and control of each defense device.
- Not permitted to make modifications to any device, only to determine what access level has been attained if any.

- Permitted to run DoS attacks against perimeter devices, but not against internal hosts.

Testing tools:

Network Associates' CyberCop Scanner 5.5, and AG Group's EtherPeek 4.0.2will be used for this assessment. EtherPeek is primarily a network monitor and sniffer with utilities for analyzing, modifying and retransmitting saved or crafted packets. It also has quick and easy to use tools for port scans, ping scans, and service scans.

CyberCop is a more robust vulnerability analysis tool with built-in modules of pre-developed tests for known vulnerabilities. Different modules can be selected based on the type of network hosts being tested. Reporting functions are also a great feature of this product as it maintains a history of all scans run and allows you to compare data from multiple scans and run queries on that data.

As mentioned above, both CyberCop and EtherPeek have utilities for scanning ports and vulnerabilities as well as crafting packets to send to a target system. However, depending on the scan being performed, it can be advantageous for the scanning host to be able to ping the target. Thus, they can have performance limitations for networks in which ICMP is disabled. One definite exception is that they both allow the sending of a crafted packet without first verifying that the target is alive.

Phase 1 – Port scan:

The tool used in this phase was the AG Group's EtherPeek. A port scan was initiated from each interface of the router and the external firewall. All known hosts were visible on the ports specified in the access lists, and no other hosts were detected.

Additional probes initiated from Cyber Cop were able to review the OS version of all the servers on the service network, but not of the firewall or router.

Verification of ICMP rules:
To verify connectivity to hosts, ICMP was enabled through the PIX for part of this scan, and debug was enabled using the following commands:

```
conduit permit icmp any any
debug icmp trace
```

As EtherPeek was used to scan for open ports, the following debug lines could be seen scrolling on the PIX console:
Outbound ICMP unreachable (code 3) 205.0.0.232 > 205.0.0.232 > 205.0.0.193
Outbound ICMP unreachable (code 3) 205.0.0.232 > 205.0.0.232 > 205.0.0.193
Outbound ICMP unreachable (code 3) 205.0.0.232 > 205.0.0.232 > 205.0.0.193
.... and so on as EtherPeek scanned each port, and the results of the scans were than no ports were open. When the egress access list was re-enabled to the outside interface of the internal firewall, the unreachable packet is not sent back to the source of the probe.

Phase 2 – In depth host probe:

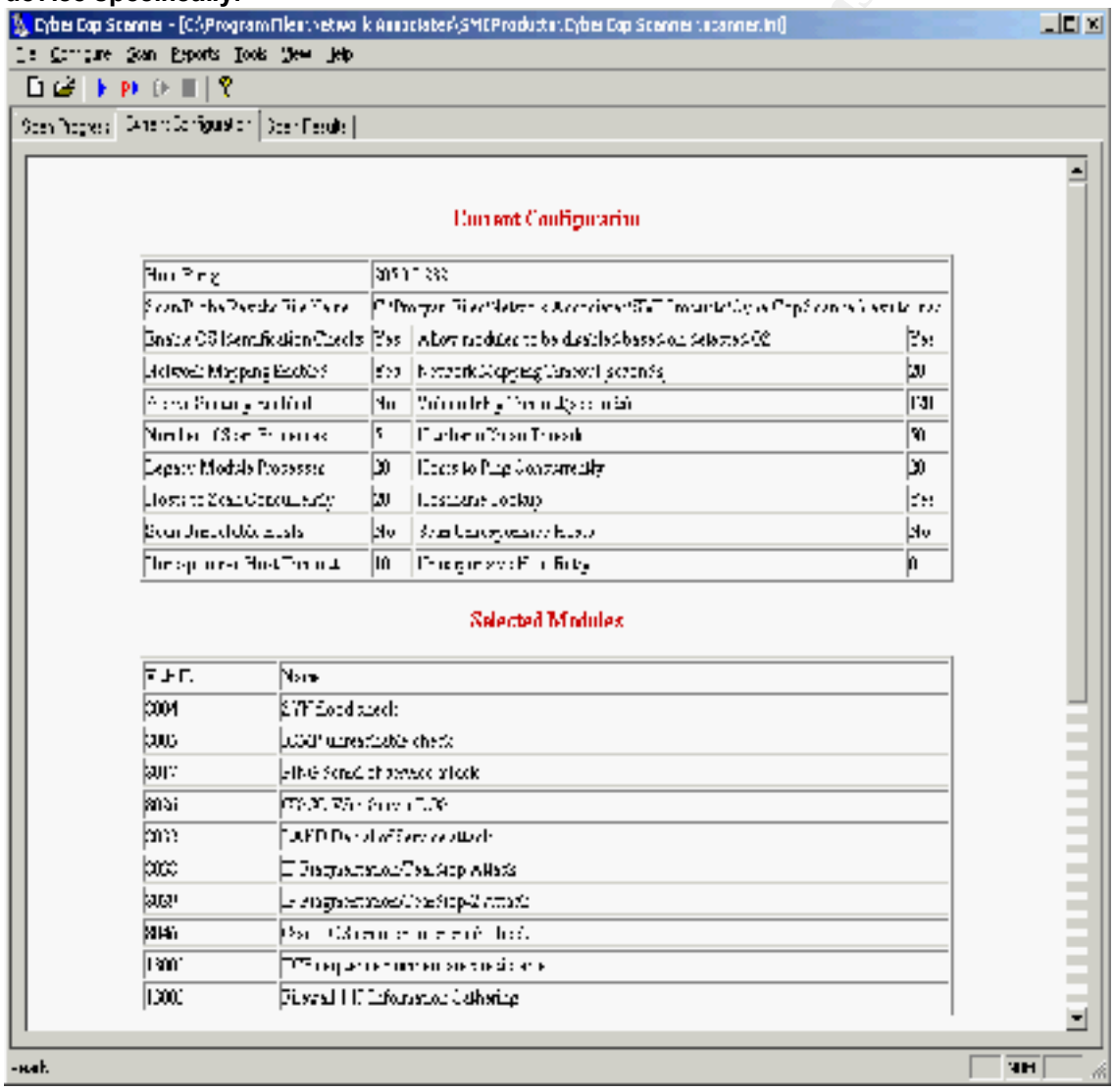
Given that the scope of this phase includes the option to attempt to cause perimeter devices to lock or crash, the Custom Audit Scripting Language (CASL) tool included in CyberCop can be utilized for the crafting of packets to be sent to a target. One script sample that's included in the software is the LAND attack which sets that source and target address to be the same as the target interface address. EtherPeek's packet crafting tool is more appropriate for making minor modification to a packet picked off the wire, and resending the new packet.

Perimeter router:
No ping allowed by design.
Not vulnerable to DoS attacks tested by CyberCop (see details below).

Primary firewall:
No ping allowed in by design.
Not vulnerable to DoS attacks tested by CyberCop (see details below).

Host scanning:
No response from the host on the vpn network.
Can only probe the web server with ports 80 and 443.
Can only probe the mail server on ports
Not vulnerable to DoS attacks tested by CyberCop.

Scans on both the router and the firewall used the following parameters set in CyberCop – except that the target address was changed in subsequent scans in order to target each device specifically:



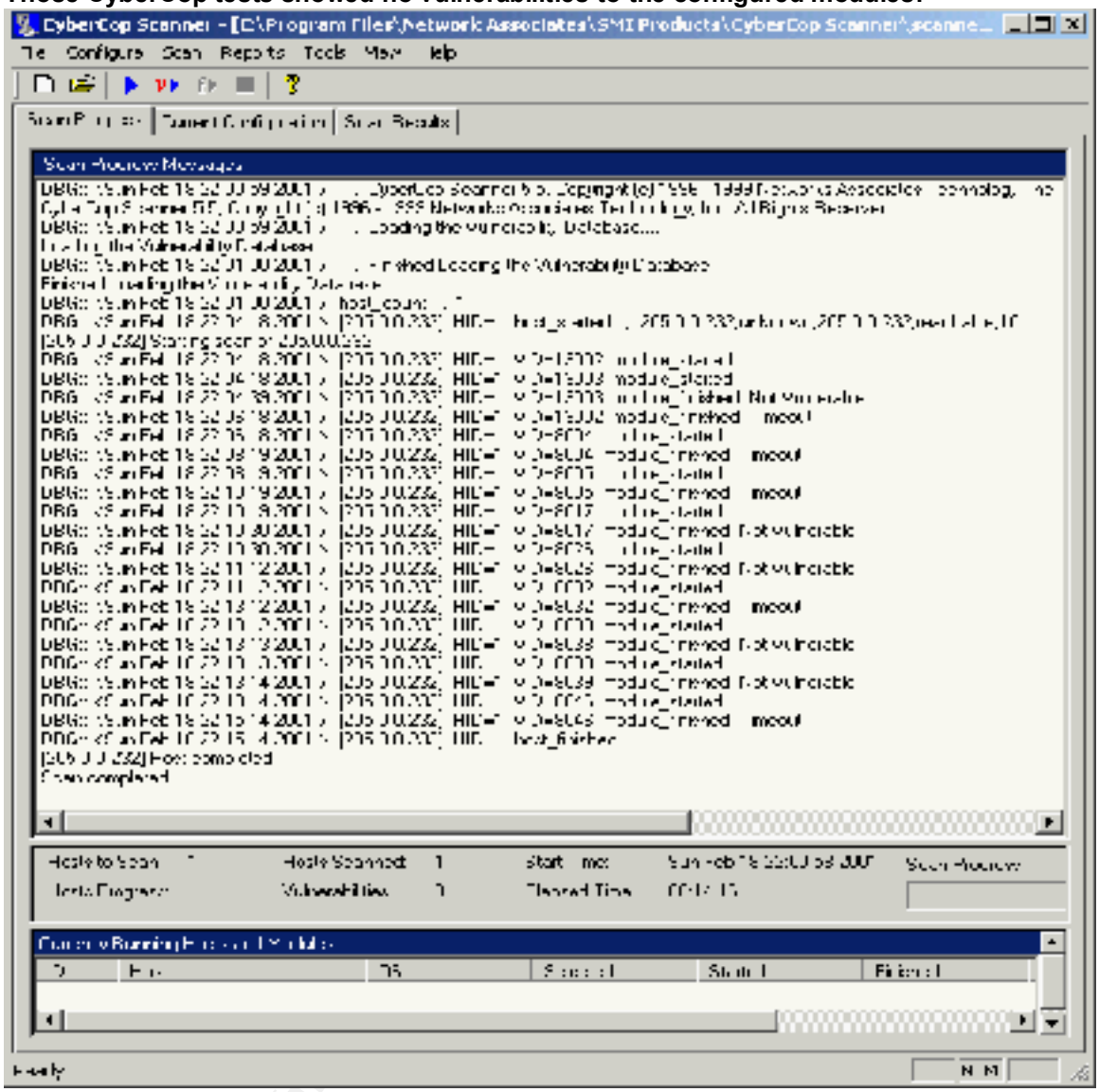
Note that these modules were chosen specifically to test vulnerabilities of routers with Cisco IOS, and some firewalls.
Some details on these particular scans:

- The LAND Denial of Service Attack sends crafted packets to the target with the target IP entered as the source IP – the source and destination ports are the same as well.

Vulnerable systems will suffer a denial of service as all resources are allocated to routing that packet in a loop.

- The IP Fragmentation Teardrop Attack and Teardrop 2 send malformed IP fragments that cause some systems to crash as they try to reassemble the fragment.
- The Cisco IOS remote router crash check sends a TCP stream to the router causing some vulnerable IOS versions to reload to the Login prompt.
- The TCP sequence number predictability test seeks to determine if hackers can guess the next sequence number in order to hijack TCP sessions.

Those CyberCop tests showed no vulnerabilities to the configured modules:



It needs to be noted that during one part of this test (and only for the purposes of testing), the following command was entered to the firewall:

conduit permit icmp any any

When ping was permitted for the sake of the scan, the internal host failed one of the tests listed above, the ICMP Unreachable Check. The conduit command was removed upon completion of the tests.

During some of the scans, enabling debug on the firewall console showed the following results:
Inbound ICMP echo request (len 32 id 3 seq 35072) 205.0.0.193 > 205.0.0.202 > 192.168.0.202
Outbound ICMP echo reply (len 32 id 3 seq 35072) 192.168.0.202 > 205.0.0.202 > 205.0.0.193
Notice the address translation in progress as the packets pass through the firewall for both the echo and the reply.

Results of scanning internal hosts:

No hosts on the DMZ were identifiable or accessible from the outside of the firewall (without the "conduit permit icmp any any" command in place).
No hosts responded to ping or any port scans (except as noted it the special ping test above).

Three hosts on the service network revealed the following vulnerabilities to be remedied:

Host 1: Web server is running IIS 5.0 on Windows 2000.
Software Vulnerabilities:
Numerous vulnerabilities exist dating back to March 2000.
Risk = Severe Various vulnerabilities exist, including the following:

1. **"Web Server Folder Traversal" Vulnerability** - Microsoft Security bulletin (MS00-078) Quoting from the Microsoft alert at http://www.microsoft.com/technet/security/bulletin/MS00-078.asp "Due to a canonicalization error in IIS 4.0 and 5.0, a particular type of malformed URL could be used to access files and folders that lie anywhere on the logical drive that contains the web folders. This would potentially enable a malicious user who visited the web site to gain additional privileges on the machine – specifically, it could be used to gain privileges commensurate with those of a locally logged-on user. Gaining these permissions would enable the malicious user to add, change or delete data, run code already on the server, or upload new code to the server and run it."
2. **"Malformed Extension Data in URL" Vulnerability** –(MS00-030) could consume all CPU resources on the server creating a denial of service.

Software Recommendations:
Apply all related hotfixes available from Microsoft at:
http://www.microsoft.com/technet/security/current.asp?productID=15
OS Vulnerabilities:
Too numerous to mention.
OS Recommendations:
There have been at least 15 security related hotfixes released for NT 4.0 since the release of the latest service pack, SP6a.
Ensure that the OS has been hardened, including but not limited to stopping all unneeded services, securing file and registry access, renaming default administrative accounts, and creating login banners.
Part of the hardening should be to ensure that the server is running the latest service pack with all the latest applicable security hotfixes.

Host 2: Mail server running Exchange 5.5 on NT 4.0
Risk = High Could cause the server to fail.
Software Vulnerabilities:
"Malformed MIME Header" Vulnerability – Microsoft Security Bulletin (MS00-082)
More information: http://www.microsoft.com/technet/security/bulletin/MS00-082.asp
Software Recommendations:
Ensure the server is running at least SP2 to address other vulnerabilities not listed here, and…
Apply the security hotfix available at:
http://www.microsoft.com/Downloads/Release.asp?ReleaseID=25443

Host 3: DNS server running Windows 2000
Risk = Low Potential denial of service from memory leak
Software Vulnerabilities:
It's possible to cause a utilization of all memory resources over time. Bugtraq ID 2007. For more information, see http://www.securityfocus.com/.
Software Recommendations:
Ensure that Windows 2000 SP1 has been applied.

Recommendations:

Following a thorough review of the access lists, firewall policies, and network layout the following recommendations are being made:

ADMINISTRATION:

1. Require system administrators to use complex passwords on all devices and to change passwords on a regular basis and whenever an administrator leaves the company.
2. Use login banners on all perimeter devices as well as all company servers and workstations.
3. Upgrade the router to a cryptographic image that supports SSH Version 1 (for version 12.0 and later). With that in place it may be possible to also implement secure centralized management of the perimeter devices.

LOGGING:

4. Use NTP to synchronize logs
5. Use SNMP for more detailed and up-to-the-minute alerts
6. Use RADIUS/TACACS+ for validation of remote access users.
7. Either be sure to disable any web server on the router:

```
no ip http server
```

Or secure that service:

```
ip http server
ip http port 8765 [use a non-standard port]
ip http access-class <1-99> [create an access-list to protect the HTTP port]
```

8. Use router command auditing. Use AAA accounting on the router and a TACACS+ server to track all commands or a limited set of commands typed into the router. AAA accounting provides information about the EXEC shell commands for a specified privilege level that are being executed on a router.
9. Use Secure Syslog (ssyslog) which is designed to replace the syslog daemon. Ssyslog implements a cryptographic protocol that allows the remote auditing of system logs. Auditing remains possible even if an intruder gains superuser privileges in the system. The protocol guarantees that the information logged before and during the intrusion process cannot be modified without the auditor (on a remote, trusted host) noticing.

Alternative architecture recommendations:

Suggestion1:
If the budget allows for two strong firewalls such as the PIX 520, use both of them together in a fail-over mode.
Things to consider when using failover:

- Each PIX interface in a subnet will need its own IP address. Thus, if both PIXes were to have a failover PIX, then the size of the DMZ subnet may need to be increased to allow for two additional Ips (one for each secondary PIX interface).
- Interface speeds will need to be statically set (already done in this configuration anyway).
- With the PIX firewalls configured in failover mode, one has a primary and one has a secondary address for each interface that they share. The primary one is always the one that's to be used by other hosts on that network as a gateway. More specifically, the active PIX is always performing proxy arp for the primary IP address… even if the active PIX is the secondary one, when the primary has failed.

Suggestion 2:
Remove the DMZ. If the number of interfaces on the firewall limits the number of networks, consider putting the development network on the other side of one of the other networks behind a light-weight (though sufficiently secure) firewall.

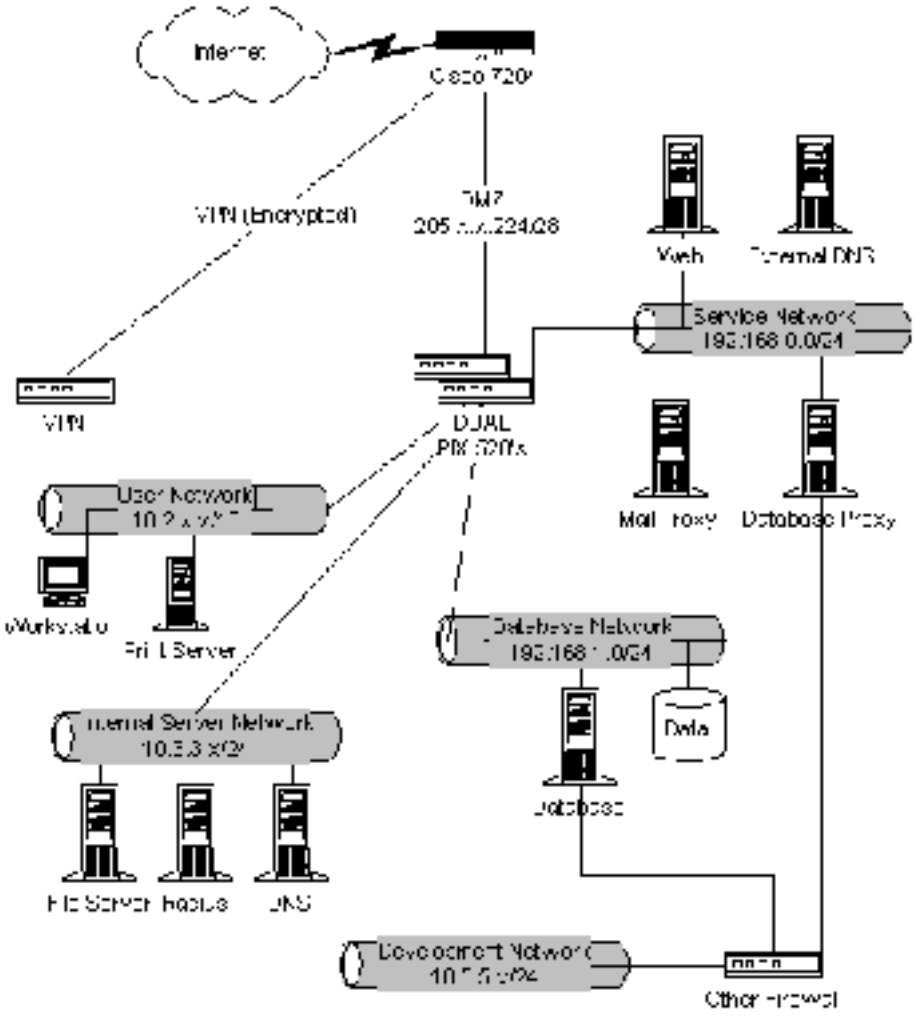
Things to consider when removing the DMZ:

- Without the DMZ (as it's currently defined in this network) the external PIX would also be the internal PIX, and thus would be performing NAT for the internal networks.
- If it was still desirable to have an IDS behind the firewall, then more IDS devices may be required to cover each network.
- More IP addresses would be needed on the "external" network to account for all the NAT addresses from the internal networks.
- Internal networks that are using NAT to talk to the outside world would still not be using NAT when communicating with each other.

If removing the DMZ is not an option and there's a concern about computers on the particular internal network sniffing traffic as it passes across to/from the development network, create a point to point tunnel between the two firewalls across the DMZ.

Suggestion 3:
Use dual-homed servers on the Service Network so their requests to database servers are no passed out over the same wire as the original client request, but instead through a secured network, preferably with another firewall between the back-end of the web server and the front-end of the database server.

The following diagram shows how the revised design would look (simplified to exclude the necessary switches and additional cabling required to support the dual PIX firewalls in a failover configuration):

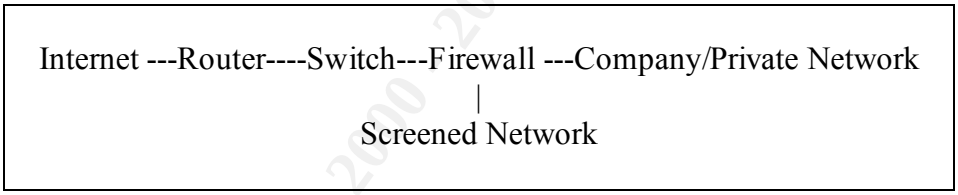


Assignment 4 - Design Under Fire (25 Points)
The purpose of this exercise is to help you think about threats to your network and therefore develop a more robust design. Keep in mind that the next certification group will be attacking your architecture!
Select a network design from any previously posted GCFW practical (http://www.sans.org/giactc/gcfw.htm) and paste the graphic into your submission. Be certain to list the URL of the practical you are using. Design the following three attacks against the architecture:

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

The practical chosen for this assignment is by Antoinnette Binning and is available at:
http://www.sans.org/y2k/practical/Antoinnette_Binning.zip

Network diagram from the documentation:



Perimeter devices and versions used on this network:

The router is a Cisco 4500, with IOS version 11.2.
The firewall is a Checkpoint Firewall-1, V4.0, with SP7.

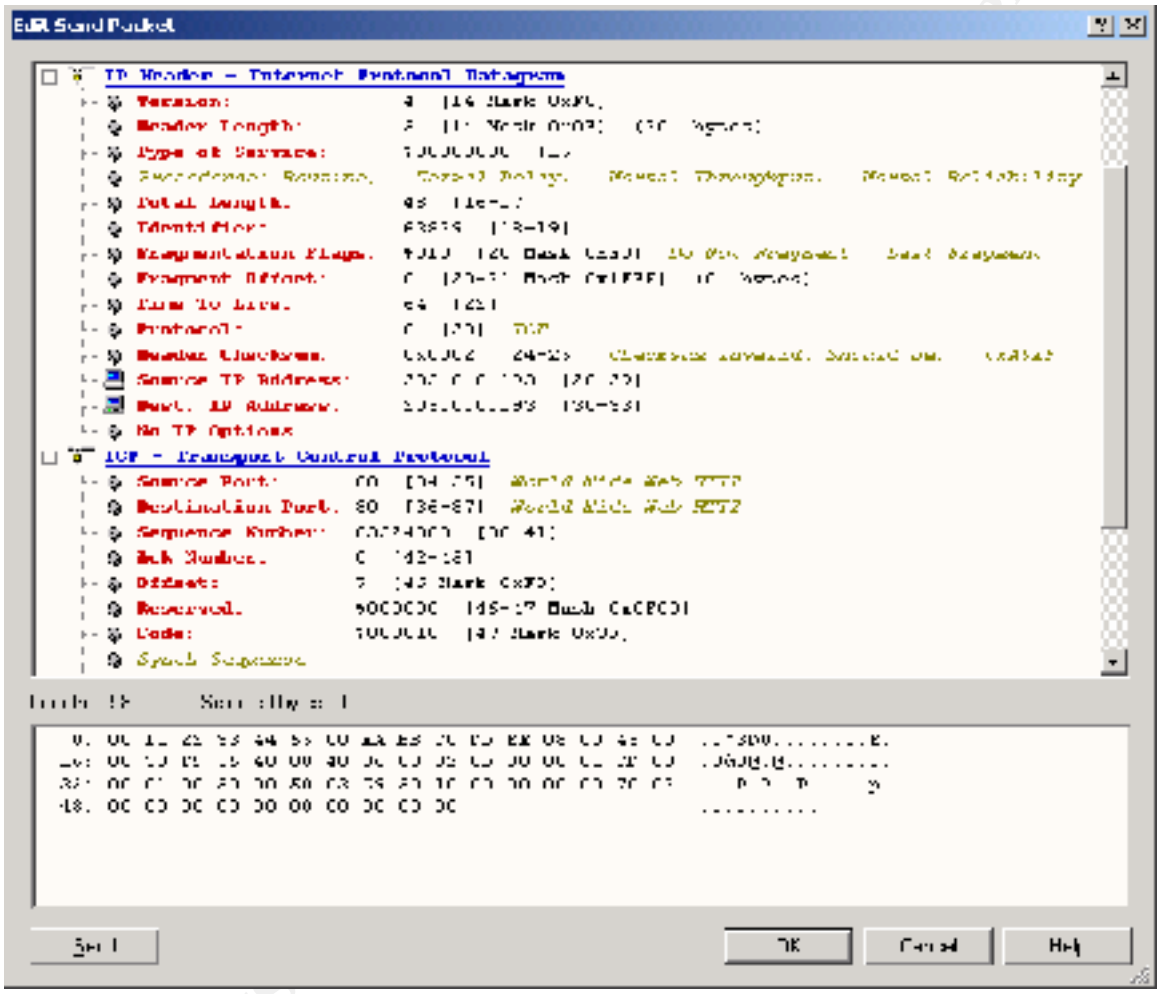
Some known vulnerabilities of FW-1 V4.0 from which this design is already proactively protected:

Attempted connection to internal rsh host.
This can only be done if the FireWall-1 administrator specifically enabled RSH/REXEC with stderr-port support in the Properties window. In this case, it's not clear from the documentation whether the reverse rsh/rexec option has been specifically disabled, but there is a firewall filter mentioned which would block all r-type traffic regardless. This is bugtraq ID 1534.

Single packet DoS
This DoS attack has not been confirmed by Check Point, so it may not really exist. Regardless, this network is protected from such spoof attacks. This suggested vulnerability would send crafted packet to the firewall with the source address equal to the destination address of the firewall interface. Firewall-1 has a built-in anti-spoofing mechanism which needs to be enabled to avoid this problem. This is bugtraq ID 1419.

It's not clear from the documentation whether the firewall's anti-spoofing mechanism is enabled, but the ingress filter on the router should mitigate this risk anyway. The ingress filter blocks internal and private source addresses from reaching the network from the outside.

If the ingress filter could be bypassed, a sample LAND packet such as the one pictured below might cause a vulnerable target to crash. This SYN packet is configured to have the same source and destination IP address, and the same source and destination port (HTTP)



Lock the server on which the firewall resides by exploiting its SMTP proxy service.
One reference states that this is a confirmed vulnerability on NT, but CheckPoint's web page does not mention that this is OS specific. Sending a string of binary zeros in a crafted packet to SMTP Security Server could bring CPU utilization to 100%. This network design is safe from this risk because it does not employ the SMTP Security Server. This is bugtraq id 1416, and there are service packs available for FW-1 4.0 and 4.1 to address the vulnerability of the SMTP Security Server.

Create a DoS attack:

This DoS attack is based on bugtraq ID 549. For details see www.securityfocus.com.

Taking advantage of the fact that all TCP connections are permitted OUTbound on port 80, this particular attack might be more easily performed from the inside. In some rare situations, this

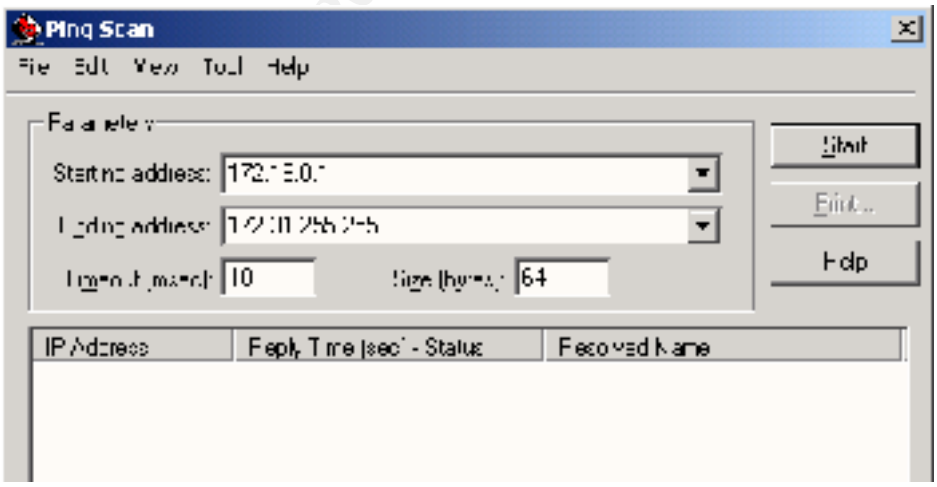
method could also be performed from the outside to a public address on a screened service/dmz network, but that would only be the case if the firewall was permitting traffic inbound to a very large number hosts that did not exist. For example if the firewall was permitting inbound HTTP to a subnet of web servers, but there was actually only one web server on that subnet.

Firewall-1 maintains a TCP connections table to verify stateful communications and return traffic. The table is typically limited in size to 25,000 to 35,000 connections - plenty for most environments. Entries in the table are cleared after one hour by default, but that timeout setting is configurable. If the TCP connections table is full, it can't assist in the establishment of any new connections until the existing entries expire from the table, thus resulting in a denial of service.

The way to fill the table is to send TCP SYN packets to a host that can't reply (e.g. one that does not exist). The way to send such an enormous number of SYN requests in a short period of time is to open hundreds of browser windows on your workstation and keep cycling through them all as quickly as possible clicking the "retry" button…. Just kidding… A network scanner would require far less effort. ;-)

Once you've located a number of addresses for invalid hosts, configure a port scanner to scan TCP at those addresses most likely on port 80. In fact, the hosts don't need to be public addresses. If you're on a 10.x.x.x network for example, you could scan for all hosts on the 172.16.x.x network, and these connections would be routed through your default gateway, the firewall. Be aware that those attempted connections could very well show up in a firewall log including the source port… thus, if secrecy is an issue, use crafted packets with an source address of an internal system that is also bogus.

This screen shot shows the parameters entered into the Ping Scan utility of EtherPeek. The address range is set to a full range of private IP address that don't exist, and the timeout has been set to 10 milliseconds from the default of 2000 (2 seconds). Modifying the timeout will let the Ping Scan run faster since it's not waiting long at all for replies before proceeding to the next address in the target range. All the packets will be sent to the default gateway for the source host… likely the firewall. If the firewall has an egress filter, it might just drop all these packets as invalid.



Create a DDoS attack from 50 remote hosts:

Locate 50 modems to use in the attack:
At your place of work, pretend you're curious about getting DSL, and ask who has what service provider… probe further by suggesting that you could communicate over the web, or even have tournament gaming sessions… and ask what there IP address is. You may not want to use your co-worker's address in the attack, but that address would give you a clue about the range of

addresses given to DSL customers by that DSL provider. Then run basic ping scans to those address ranges to see which ones are on-line most of the time.

It's quite unlikely that those modems or the ISP are logging ping activity since it's so common, so the attack may be difficult to trace. Regardless, you might not want to originate the attack from your home or work computer unless you want to get caught.

To initiate the attack, send a stream of crafted ICMP ping packets to those 50 modems with the return address of the target system. This will cause all 50 modems to send a stream of ICMP replies to the target. The router on the target network specifically denies incoming Echo packets, but it still permits Echo-reply. The router itself would be a likely target.

The firewall may be more resilient to this type of flood, but there may be a public host on the screened network behind the firewall that is vulnerable. To exploit such holes, you may want to set the source address of the crafted echo packet to be an address of the web server or mail server. While you're at it, try setting the source port (i.e. the destination port of the packet sent from the DSL modem) to port 25 or 80.

To protect from this type of ICMP flood attack, the following steps can be taken:

- Echo replies from the Internet could be blocked entirely.
- Echo replies can be permitted during limited testing periods, then denied again.
- Echo replies can be permitted only to certain hosts known to be resistant to ICMP floods.

Target an internal host from the outside:

The mail server is being chosen here as the internal target because it's known to exist, known to be critical to any business, and known to have functionality and information that would be interesting to exploit.

If the intent was to disable the mail server, it could be targeted in DDoS attack similar to the one in the previous example. The address of the public mail server could be entered as the targeted for an ICMP flood attack. Additionally, that mail server could be subject to a TCP flood on port 25.

However, the exploitation suggested here is a less destructive, and has less obvious (if not completely undetectable) results. This approach will attempt to use some of the features of the SMTP protocol and mail server to gather information about users on the target network. This target network has no controls in place on the firewall to filter the types of commands that could be sent to the mail server. Also, as noted in the documentation, port 25 is open from any source to the mail server.

The server is listening on port 25, and the firewall is allowing through communications on port 25 to that server. That doesn't mean that we need to be sending mail. We could configure a telnet program to open a command line session with the mail server on port 25, and allow us to enter commands. Many servers may be resistant to this type of reconnaissance by restricting the type of access, or requiring authenticated access, but it's worth a try.

Some commands that could be used to gather data from this server are EXPN and VRFY. EXPN tells the server to expand a distribution list of users including their names and email addresses. VRFY is used to verify that a name actually exists in the address list.

The following is a sample list of send and reply lines for the VRFY command straight from the SMTP RFC 821 http://www.ietf.org/rfc/rfc0821.txt?number=821:

"S: VRFY Smith

R: 250 Fred Smith
Or
S: VRFY Smith
R: 251 User not local; will forward to
Or
S: VRFY Jones
R: 550 String does not match anything.
Or
S: VRFY Jones
R: 551 User not local; please try
Or
S: VRFY Gourzenkyinplatz
R: 553 User ambiguous."

Also from RFC 821, here are some examples of using the EXPN command:

" S: EXPN Example-People
R: 250-Jon Postel
R: 250-Fred Fonebone
R: 250-Sam Q. Smith
R: 250-Quincy Smith
R: 250-
R: 250
Or
S: EXPN Executive-Washroom-List
R: 550 Access Denied to You."

To alleviate this concern, the SMTP Security Server could be enabled on the firewall. The proxy server could analyze and drop packets with invalid commands. However, that service can make the firewall itself vulnerable to DoS attacks: Sending a string of binary zeros to port 25 on the SMTP Security Server can bring CPU utilization on the server to 100%. This is bugtraq id 1416, and there are service packs available for FW-1 4.0 and 4.1.

Sources and reference materials for all assignments:

GCFW training materials from Capitol SANS, December 2000

SANS web site
www.Sans.org

Cisco's web site
http://www.cisco.com

Designing Network Security by Merike Kaeo, Cisco Press – 1999

Advanced Cisco Routing Configuraiton by Laura Chappell, Cisco Press – 1999

Building Cisco Remote Access Networks by Catherine Paquet, Cisco Press - 1999

Security Focus
http://www.securityfocus.com

Check Point
http://www.checkpoint.com/techsupport/alerts

Real Secure
http://www.realsecure.com/

Cyber Cop
http://www.nai.com/

Common Vulnerabilities and Exposures
http://Cve.mitre.org

Black Hat
http://www.dataprotect.com/bh2000/