Global Information Assurance Certification Paper

GIAC SANS GCFW Practical Assignment
Firewalls, Perimeter Protection and VPNs

version 1.5e

Christopher M. Kellogg, chris.k004
Lone Star SANS II
Original Submission

Author's Note

I have had restricted access to resources for the purposes of assembling and testing the environment I have proposed. As a result, there may be minor limitations in the quantity of documentation I am able to produce with respect to screenshots, application output and hardware testing. Configuration, testing and auditing was done whenever possible with a Cisco PIX515, a production Nokia IP440/CheckPoint Firewall-1 device, and a production Cisco VPN Concentrator 3015. Some of the documentation and application output appearing in this practical was written without performing the tests or running the applications as indicated. Rather than be study in the precise stimulus/response of security hardware, or as a line-by-line security configuration how-to, this paper has been written to show a level of competence, knowledge and experience with the concepts and tools used in the realm of network security.

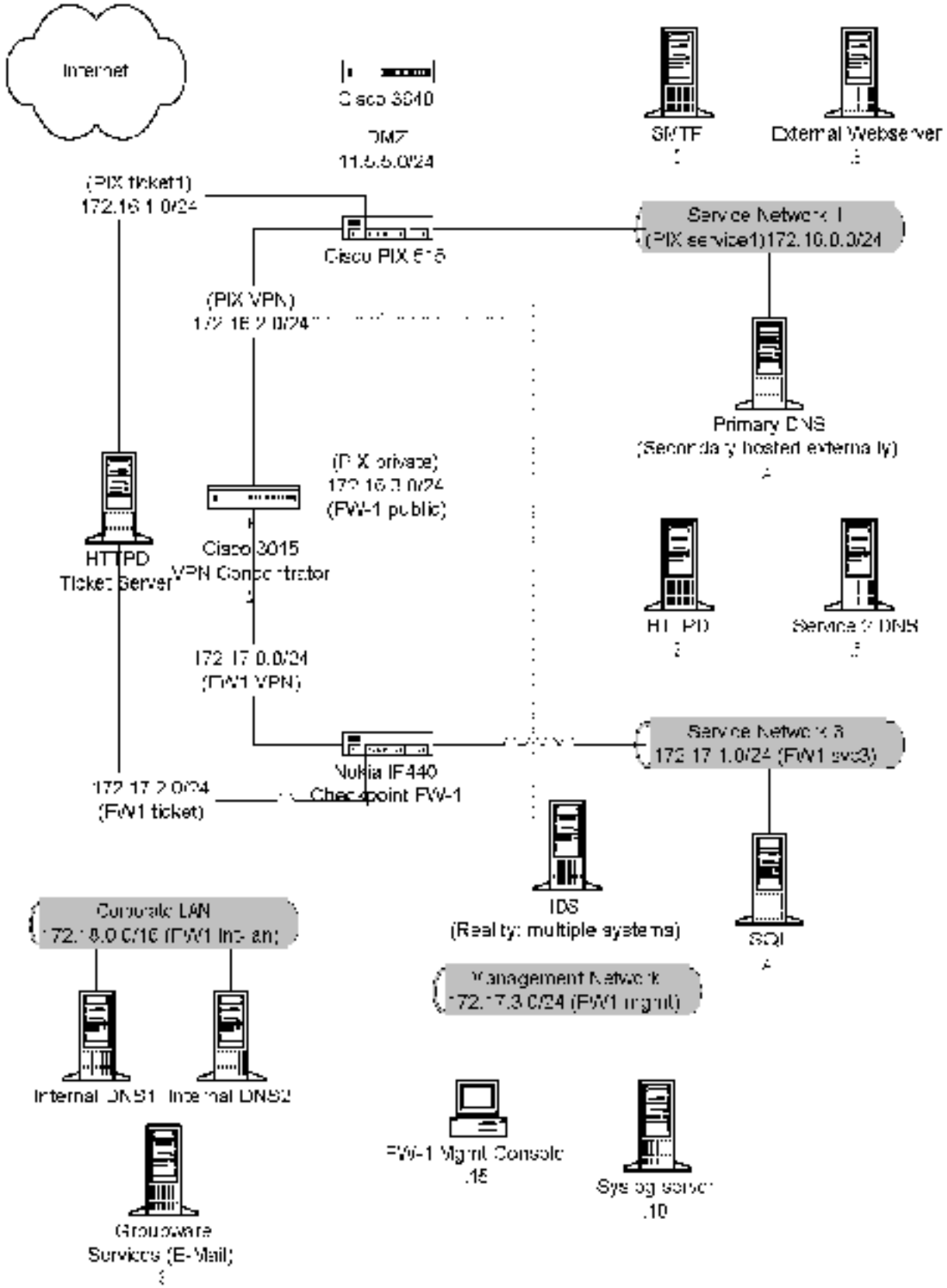
A. Security Architecture

A.1. Network Design

One of the most basic parts of any corporation's security backbone is the physical distribution, configuration, and connection of basic network hardware. The proper usage of routers, switches, firewalls and IDS can help limit the risks involved when utilizing direct Internet connections. Conversely, even the best firewall/router configuration could be quickly bypassed if the network were simply thrown together without taking into account a solid security configuration.

A network with multiple network entry points is too much of a risk, outside any requirements for redundancy and load balancing. Placing a VPN device in parallel with a well configured firewall is effectively opening a back door to the network, giving potential attackers another option when attempting to gain access to a multi-million dollar e-commerce company's infrastructure. It is possible to shield some, if not all, of a company's information resources behind a device designed to filter out unwanted traffic with limited effect on more peaceful traffic. Naturally there are methods to abuse any Internet service, however those attacks would be severely limited in scope with this proper network design.

Given the fact that GIAC Enterprises is planning on doing a great deal of business over the public internet using varied methods of communication, a fairly robust network model is warranted.



Public Internet traffic will enter through a Cisco 3640 router, configured to filter out the more prevalent attacks and scans at the perimeter. These attacks and scans are blocked with no response, giving as little information to the attacker as possible. Additionally, this device will perform a small amount of egress filtering to prevent private IP spoofing, limiting the value of this network for hackers. While this is a good start, a router is not the optimal device for filtering, and therefore will be using only a basic filter configuration; a large configuration could potentially cause a noticeable slowdown in throughput.

The 3640 establishes a DMZ where the primary firewall is found. This firewall, a Cisco PIX firewall device configured with 5 network interfaces in use and an unused sixth interface (which could be used for honeypots, perhaps). Public Internet traffic will be filtered and passed to the appropriate service network dependent on the type of traffic, destination, and the state of the connection the traffic was passed as a part of. The choice of a Cisco PIX515 was due to the basic design premise of 'block first, ask later', for the relatively limited number of weaknesses reported in the past, and also partially due to the fact it keeps no secrets and tells no lies – if it's configured to deny, it will deny without question.

Attached to the firewall's second interface is the primary service network, where the primary HTTP, SMTP, and DNS servers are located. This design uses a mixture of server operating systems mostly for performance and ease of configuration aspects: Windows 2000 is used for the HTTP server, RedHat Linux 7.1 is used for the DNS and SMTP relay servers and intrusion detection services. Should FTP services be required, SSH will be installed on one of the Red Hat Linux servers and SSH/SCP will be used to securely transmit files, user accounts and passwords. Windows 2000 has good ease-of-use, although it can be at risk of attack due to the name behind the product; a good update policy as discussed later will reduce this risk. Red Hat Linux is a solid product, again at slight risk of attack due to its relative popularity and name recognition; updates will also reduce this risk as will a limited scope of configured functions.

On the second service network, using the third interface in the PIX, is the entry point for VPN connections. A Cisco 3015 hardware VPN device provides a solid solution for both remote clients and network-to-network VPN connectivity, while placing it behind a primary firewall limits the ability of hackers to exploit any weaknesses that might appear in the base operating system of the server. Traffic passed to the 3015 will be only logged for start and end of session, as the traffic itself will be encrypted and unreadable on the public side of the box. Traffic passed to the Cisco box will be limited to ESP and ISAKMP TCP ports.

The fourth network connection of the PIX will be sent to a service network leading to a second firewall: a Nokia IP440 network appliance with CheckPoint Firewall-1 equipped with six interfaces. The network between the two firewalls will carry SMTP-relaying traffic as well as all outgoing Internet requests from the MGMT network and corporate LAN, both discussed later in this document.

The final network connection on the PIX is used as an interface for database requests from the primary service network. In order to best protect the database services, all requests will be formed using a ticket authentication system passing through a second, shielded, web server. The ticket will contain basic authentication and checksum information as well as the information

request. The ticket web server will verify the ticket, check for appropriate rights, and if all information can be internally authenticated it will form and send a database request to the appropriate database server, returning the results to the requesting server. This prevents the automatic takeover of a database server when a webserver is compromised.

The second firewall has three interfaces used by previously mentioned connections (VPN, WWW-ticket server and the service network leading to the outside firewall). This FW-1 device also has 3 additional networks: The corporate LAN, a management network with firewall management application and other security management services, and the application service network. The application service network contains the essential data for corporate functions, made available only to traffic sourced from the LAN, MGMT and the VPN exit service network. Using the same ticket concept from the outside server, a WWW server is used here for VPN-ed connections from employees, remote sites and business partners (distributors, suppliers, etc.).

A.2. Intrusion Detection

A solid network design and implementation can go a long way in protecting mission-critical servers and data from the typical attacker. The multiple layers of firewalling and filtering could even stop a more determined attacker. In the end, however, the best security policy is one monitored by protection systems, which alert administrators in the case of a serious attack or breach. Firewalls are designed to stop novice attackers; they will only slow capable and determined attacks. As an attacker fights with these security devices, the IDS will detect the intrusion and notify administrators who can take appropriate measures to shut down the attack.

An IDS workstation is placed on every access network to monitor traffic for questionable activity. Each IDS will be equipped with 2 network interfaces: The first is normally configured and attached to the MGMT network for logging and remote access purposes; the second is semi-configured, enabled but does not have an IP address. It is configured in promiscuous mode, gathering all traffic as it passes and logging all activity using the Snort basic IDS application. Although the IDS itself could become a target for attack, the likelihood of such an attack being successful is fairly limited given the system configuration if the monitoring applications are kept up-to-date.
Note: The network diagram shows only one IDS, this is simply done for brevity, each connection is a separate IDS device.

The syslogd server is on the MGMT network. While allowing the passage of syslog information to this network constitutes a minor security risk, watching for and patching new weaknesses in syslog will ensure a limited vulnerability. The IDS systems also connect to the MGMT network and send their captures to the syslogd server.

Swatch is run on the log server, giving the administrators notification when certain predetermined criteria are met. Swatch will be configured with several levels of notification dependent on the type of event logged. A service restarting may only draw passing interest and perhaps notification via e-mail, a root login or 'su' attempt may be of more concern. In all, this connectivity and layout gives the syslogd server access to network resources, such as e-mail, and also gives administrators the ability to view logs from their desktop using SSH.

A.3. Remote Access/VPN

GIAC Enterprises is going to do most of it's business online creating a significant need for customers, suppliers, and partners to gain access to certain resources the company has developed. The most common user is the customer, who will access data on the primary webserver in the first service network using SSL/HTTPS. Using a browser with 128 bit key encryption and a certificate issued by one of the public certificate corporations (Verisign, Thawte or the like), transactions can be made with an acceptable level of risk.

Suppliers and Partners are a little more interesting. Using a VPN client connecting to the Cisco 3015 VPN Concentrator, user accounts for each location set with strong passwords, and assigned network list and filters, it is possible to quickly and efficiently provide access to users without a serious compromise in security. Split tunneling will be disabled to protect resources from the potential of an attack using the VPN client as an entry point. The use of ESP versus AH (partially due to hardware constraints) will give greater flexibility to the placement and configuration of VPN clients.

A.4. Updating, Patching, and Awareness

To help best utilize this network design and IDS implementation, a solid and consistent plan to update and patch server software must also be in place. New security loopholes are found every day; daily checks of information sources are a necessary part of any complete security plan. Without proper patches, any machine would be a target for attacks in the event of a new vulnerability or even older vulnerabilities that have been exposed and corrected. There are several sites dedicated to system security, some of which are listed here:

- http://www.microsoft.com/windows2000/downloads/critical/default.asp
- http://www.microsoft.com/technet/security/default.asp
- http://www.windowssecurity.net/
- http://www.securityfocus.com
- http://www.redhat.com/support/errata/rh71-errata-security.html
- http://www.linuxsecurity.org/
- http://www.checkpoint.com/

The networking and VPN hardware is also susceptible to weaknesses in the low-level operating system installed in each respective device. While these devices may have some internal protection scheme to limit remote access either through built-in controls or access lists, new issues are found for them often as well. The following sites are useful in keeping up to date:

- http://www.cert.org
- http://www.cisco.com

B. System/Device Configuration

B.1. Border Router Configuration

Note: The complete Border Router ACL is found in Appendix A.

The border router is nothing more than a simple router. It has been designed for maximum throughput with limited delay, and is not necessarily the best device to use for extensive filtering. The router is also the first device any incoming traffic will encounter.

There are some tradeoffs that can be made to minimize latency while still utilizing some basic packet filtering on the router. To make sure there is as little impact as possible, the quantity and scope of rules used on the border router will be limited to the more prevalent packets it would be best to disrupt.

Before the creation of the ACLs designed to restrict unwanted packets, the router itself needs to be locked down from unwanted attacks and connections. The router will also be configured to not send ICMP admin replies to rejected packets and ensure other basic services are disabled.

```
no ip unreachables
no ip direct-broadcast
no service tcp-small-servers
no service udp-small-servers
no service finger
no ip http server
no ip bootp server
no snmp
no ip source-route
```

The device will block all private IP sourced packets as well as falsified 127.0.0.0 addresses.

```
access-list 101 deny ip 192.168.0.0 0.0.255.255
access-list 101 deny ip 172.16.0.0 0.31.255.255
access-list 101 deny ip 10.0.0.0 0.0.0.255
access-list 101 deny ip 12.0.0.0 0.255.255.255
```

Order is important for performance concerns when dealing with ACLs on border routers, and this router will be handling the traffic that is the lifeblood of the company. Latency is an issue, and causing it at this level will have a notable impact at the user's desktop. For this reason, there is a rule at the top of the rules to allow legitmate traffic to bypass the remainder.

```
access-list 101 permit tcp any 11.5.5.3 eq 80
access-list 101 permit tcp any 11.5.5.3 eq 443
```

Note: 11.5.5.3 is used as an example, in this case the IP of a public http/https server.

This design includes a significant number of Cisco devices so the border router will block several well-known Cisco protocols to limit a product-line attack.

```
access-list 101 deny tcp any any eq 49
access-list 101 deny udp any any eq 49
access-list 101 deny tcp any any eq 130
access-list 101 deny tcp any any eq 131
access-list 101 deny tcp any any eq 132
access-list 101 deny tcp any any eq 1467
access-list 101 deny tcp any any eq 1741
```

Now to block the more common attacks as found on http://www.incidents.org/ Top-Ten lists. Using basic ACLs on the router, it is possible to stop these attacks at the border and perhaps give the admins a little breathing room should a new weakness become known before a patch can be made available. A good example is portmap/SunRPC:

```
access-list 101 deny tcp any any eq 111
```

The router will also block the Top-Ten attacker IPs much like above, as done for the top two in this example:

```
access-list 101 deny ip 210.72.224.240 0.0.0.0 any
access-list 101 deny ip 64.166.1.197 0.0.0.0 any
```

Paying attention to the Top-Ten growing ports scanned is a good way to see a new attack or weakness before it falls in the hands of the general public. Ports that see an increased level of activity should be blocked in the event the weakness the scans are probing for exists on local devices.

Depending on the amount of filtering the border router is to perform, it might be useful to block additional services based on known exploits and weaknesses. Simply replying on a known exploitable port, or even not explicitly denying on that port, could spark additional interest it would be best to. In this example, the route will block all traffic destined for the NetBIOS ports:

```
access-list 101 deny tcp any any eq 137
access-list 101 deny udp any any eq 137
access-list 101 deny tcp any any eq 138
access-list 101 deny udp any any eq 138
access-list 101 deny tcp any any eq 139
access-list 101 deny udp any any eq 139
```

Now to block the standard ports it would be best to reject traffic on without regard, perhaps BackOrifice or telnet. The danger of the attack as has to be weighed against the restriction on services; bear in mind that high-range ports are also used for PAT connections; blocking too many could result in failed sessions for users inside the firewall.

```
access-list 101 deny tcp any any eq 23
access-list 101 deny tcp any any eq 31337
```

```
access-list 101 deny udp any any eq 31337
```
Note: Use http://www.sans.org/newlook/resources/IDFAQ/oddports.html for more candidates.

Finally, since the router is blocking ports on an individual basis it also has an explicit deny at the end of the ACL. To ensure any necessary traffic that hasn't been dealt with so far is able to arrive at it's destination, there are two more lines to add to the ACL.

```
access-list 101 permit tcp any any
access-list 101 permit udp any any
```

With a functional ACL for inbound Internet traffic, it is time to consider what traffic is sent back out to the Internet. There is a fairly long list of inbound rules, so in order to keep the performance level high this filter will only stop the outgoing use of private IP addresses.

```
access-list 102 deny 10.0.0.0 any 0.255.255.255
access-list 102 deny 172.16.0.0 any 0.31.255.255
access-list 102 deny 192.168.0.0 any 0.0.255.255
access-list 102 deny 127.0.0.0 any 0.255.255.255
```

With a pair of complete ACLs, it is time to them into use. The following configuration entries will place each ruleset on the appropriate interface.

```
ena
conf t
inter ser0
ip access-group 101 in
exit
inter eth0
ip access-group 102 in
exit
exit
wr mem
```

In the end, these rules are designed to protect the valued network devices and applications without overburdening non-specialized resources with commands and configurations that could impact the traffic needed to do business and survive. To see the complete border router ruleset, please go to Appendix A.

B.2. Primary Firewall

The primary firewall is where the first harsh action is taken to provide protection to the valuable Internet resources on the network. This device will act both as a shield and to limit the value of the network for the purposes of relaying or spoofing attacks. The weapon of choice is the Cisco PIX 515, a device that is capable of expanding to the needs of the company and providing a high level of availability. Since it is a Cisco device the configuration is similar to the 3640 border

router configuration in the previous section. The major difference being that the 515 is designed to sort and filter packets in contrast to a router that passes them along as efficiently as possible..

After configuration, this particular PIX has 5 interfaces in use, each earmarked for specialized purposes. With this configuration, it is possible lock down interfaces so as to prevent access from point to point without following a defined method. Since the logging on a PIX firewall is valuable when using a syslog server, one will be placed on a later network to ensure that information can be captured and viewed.

First to provide the basic information that determines the capability of each interface. In order to make viewing and interpreting the information easier, each interface will be named according to its approximate function.

```
nameif ethernet0 public security0
nameif ethernet1 service1 security10
nameif ethernet2 ticket1 security15
nameif ethernet3 vpn security20
nameif ethernet4 private1 security25
nameif ethernet5 service2 security5
```

The PIX firewall uses security to determine the level of access a device has to the next. A higher level of security, as determined by the number assigned to the interface, can be quickly configured with access to a lower security interface. There will be some cross-interface access to certain servers and devices, but such will be limited to ensure a high level of security for all services and applications.

With the interfaces named, the next step is to assign IP addresses to them. Each server, workstation, and device will have private IPs that will be NATed by the 515. This design uses the public address space 11.5.5.0/24; the private network range will be 172.16.0.0/11 for the internal (private) IP range.

```
ip address public 11.5.5.2 255.255.255.0
ip address service1 172.16.0.1 255.255.255.0
ip address ticket1 172.16.1.1 255.255.255.0
ip address vpn 172.16.2.1 255.255.255.0
ip address private1 172.17.0.1 255.255.255.0
ip address service2 172.16.3.1 255.255.255.0
```

The PIX will need to log all connections and events to the local syslogd, located on a network behind the second level firewall. The PIX has several levels of logging; it will be configured to use the 'warning' level to catch all events of level 4 and lower (a lower number indicating higher level of importance). In the event more logging is needed, perhaps to track questionable activity, the 'notification' or 'informational' levels could be used.

```
logging host private 172.17.3.10
logging trap warnings
logging timestamp
```

```
logging on
```

The next step is to allow internal users the capability to start connections on lower security interfaces. The NAT command is used to specify the source address, in this case the second firewall. In this process, PAT will also be put in place to give internal users access to Internet and service network resources. This PIX is configured to not give user access to the VPN or Ticket service networks, as these devices have specific jobs that do not require or desire access from LAN users. Additionally, it is set to restrict them to only the application ports they will need.

```
nat (private1) 25 172.17.0.2 255.255.255.255
global (public) 11.5.5.254 netmask 255.255.255.0
global (service1) 172.16.0.254 netmask 255.255.255.0
```

The local users will not need unfettered access to the public Internet or service1 networks. The next firewall inwards blocks unnecessary traffic, but it's best to be redundant in the case an internal user tries to attack from the inside out. Order is of less importance here, since the entire list, as well as other applicable lists, are parsed before passing/failing a packet.

First to protect the first service network:

```
outbound 10 deny 172.18.0.0 255.255.0.0 any any
outbound 10 permit 172.18.0.0 255.255.0.0 smtp tcp
apply (service1) 10 outgoing_src
```

Then limit user's Internet applications:

```
outbound 20 deny 172.16.0.0 255.224.0.0 any any
outbound 20 permit 172.18.0.0 http tcp
outbound 20 permit 172.18.0.0 443 tcp
outbound 20 permit 172.18.0.0 ftp tcp
outbound 20 permit 172.18.0.0 ssh tcp
outbound 20 permit 172.18.0.0 53 tcp
outbound 20 permit 172.18.0.0 53 udp
apply (public) 20 outgoing_src
```

The firewall is next configured to give the service1 network public access servers a functional connection to the Internet. The first statement is to establish a static NAT mapping from a private IP to a public IP, then the commands move on to determine which ports will be forwarded to the server by building a "conduit", or pipe through the firewall. If the configuration isn't carefully thought out, the firewall will end up with a Swiss cheese effect and it's effectiveness and value will be reduced. The PIX515 blocks all traffic by default, and these rules are only making pinholes to allow necessary traffic.

```
static (service1,outside) 11.5.5.3 172.16.0.3 netmask
255.255.255.255
conduit permit tcp host 11.5.5.3 eq www any
conduit permit tcp host 11.5.5.3 eq 443 any
```

```
static (service1,outside) 11.5.5.4 172.16.0.4 netmask
255.255.255.255
conduit permit tcp host 11.5.5.4 eq domain any
conduit permit udp host 11.5.5.4 eq domain any
static (service1,outside) 11.5.5.5 172.16.0.5 netmask
255.255.255.255
conduit permit tcp host 11.5.5.5 eq smtp any
static (vpn,outside) 11.5.5.6 172.16.2.2 netmask 255.255.255.255
conduit permit tcp host 11.5.5.6 eq isakmp any
conduit permit tcp host 11.5.5.6 eq esp any
```

The ticket server will also need to be available to the HTTPD in the service network. Since the connection is stateful, i.e. no data is sent unsolicited from the ticket server to the webserver, there only needs to be a single direction static map and conduit.

```
static (ticket1,service1) 172.16.0.253 172.16.1.2 netmask
255.255.255.255
conduit permit tcp host 172.16.0.253 eq www any
```

B.3. VPN

Remote users, distributors and partners will gain access to applications through a secure, 3DES virtual private network using a Cisco 3015 concentrator. The 3015 currently supports NATed workstations making it a very attractive option for working with clients behind a firewall. Additionally, it supports load balancing and fail-over, leaving the door open to improve quality of service should the need arise.

The configuration of the 3015 is fairly straightforward on a data protection level. The 3015 does not support AH, which makes any comparison between AH and ESP strictly academic. The VPN device will be configured to use the ESP/IKE-3DES-MD5 security association. 3DES 168-bit encryption will be used for VPN data; IPSec will use ESP with HMAC-128 encryption and MD5 hashes for authentication; and as required by the IPSec standard, authentication key exchanges will use HMAC-128 bit encryption with MD5 hash authentication

Comparing the two IPSec authentication/encryption options, ESP uses packet sequence numbers to protect against a replay attack and can pad packets to provide a limited level of data-quantity spoofing; AH can provide a higher level of authenticity due to the fact that it includes the IP header as part of the packet verification process.

An advantage of ESP in the real world is the handling of Network Address Translation (NAT) or Port Address Translation (PAT) clients; since the IP header is not included in the verification process it is possible for intermediary devices to modify the header as needed to allow for NAT/PAT. This allows the optimal level of compatibility when firewalling and NATing are the norm.

More complicated is the user/group configuration. Since there will be a variety of users accessing the VPN concentrator, it is necessary to design groups that can provide/restrict access appropriately. The Cisco 3015 can delegate access by group, each group can be assigned a series of rules and filters limiting or allowing the passage of packet depending on the packet type and destination. In the case of a typical partner, access will be granted to allow traffic only to the application webserver.

The first step in the device configuration is building the network lists. There are three user types coming into the network via VPN: The partner/distributor, the road warrior employee/work-at-home type, and the security/network administrators. The Cisco 3015 makes the configuration job straightforward, as it only requires the assembly of a few network lists. Given three user types, there will three network lists:

List 1: **Application** (Web-based applications and DNS)
Includes only the IP address for the webserver handling the application.

List 2: **Employees** (Groupware (E-Mail) Services with Application)
Includes the IP address for the Groupware server.

List 3: **MGMT** (Admin/Management Services with Employee and Application)
IP Addresses for the administration and monitoring servers in the MGMT network.

With the network lists complete, rules are created to permit access to application types within each network. The rules are built to specify what types of access are going to be either explicitly allowed, or explicitly denied. There will be very few applications so these rules will be to allow, or forward, connections and the default will be to drop. This configuration assumes the source to be the VPN client, 'to' traffic is out from client to server, 'from' is the reverse. A group of these rules is assigned to a filter, which is then applied to a Group to forward the required connections and drop the remainder.
Note: Screenshots from building the Application security list can be found in Appendix C.

Application: Allow traffic to 172.17.1.2 port 80 and port 443 (TCP)
Allow traffic from 172.17.1.2 on any port (TCP)
Allow traffic to 172.17.1.5 port 53 (TCP/UDP)
Allow traffic from 172.17.1.5 on any port (TCP/UDP)

Employees: Allow traffic to 172.17.1.2 port 80 and port 443 (TCP)
Allow traffic from 172.17.1.2 from any port.
Allow traffic to 172.17.1.5 port 53 (TCP/UDP)
Allow traffic from 172.17.1.5 on any port (TCP/UDP)
Allow traffic to 172.18.1.3 port 80 and port 443 (TCP)
Allow traffic from: 172.18.1.3 on any port (TCP/UDP)

MGMT: Allow traffic to 172.17.1.2 port 80 and port 443.
Allow traffic from 172.17.1.2 from any port.
Allow traffic to 172.17.1.5 port 53 (TCP/UDP)
Allow traffic from 172.17.1.5 on any port (TCP/UDP)

Allow traffic to 172.18.1.3 port 80 and port 443 (TCP)
Allow traffic from: 172.18.1.3 on any port (TCP/UDP)
Allow traffic to 172.17.3.10 on port 22
Allow traffic from 172.17.3.10 on any port.

The final aspect in maintaining network security with respect to VPNs is the client workstation. It is imperative that the systems used to VPN to the internal application servers are individually managed and monitored and that the users are educated in the method of use. Since these machines will be outside the control of GIAC Enterprises, it is also imperative that a set of guidelines be assembled and distributed with the information on accessing the VPN.

B.4. Secondary Firewall

The second firewall is a Nokia IP440 running CheckPoint Firewall-1. This firewall will end up as busy as the PIX515 since it will be dealing with VPN clients, Internet clients (through the ticket server), as well as Management and local user clients. If the PIX515 is compromised, this firewall will also need to keep intruders out of the internal networks while the admins take control of the situation.

Rules are formatted as they would be found on a true firewall in lieu of screenshots. 'Installed on' is always "FW-1", 'Time' is always "Any" and 'Comments' follow the ruleset.

No.	Source	Destination	Service	Action	Track
1	Any	FW-1	Any	Drop	Alert
2	Int-lan Mgmtnet Vpnnet	Svc3app	http https	Accept	Long
3	Ticketnet	SQLsrvr	MS-SQL	Accept	Long
4	Vpnnet Public Svc2net	Syslogd	Syslog	Accept	Long
5	VPN	Exchangesrvr	http https	Accept	Long
6	Int-lan Mgmtnet	Public	smtp http https ftp ssh dns domain-tcp domain-udp ms-rdp	Accept	Long
7	Public	Exchangesrvr	Smtp	Accept	Long
8	Vpnnet	Mgmtnet	Ssh	Accept	Long
9	Any	Any	Broadcastsvcs	Drop	
10	Any	Any	Any	Drop	Alert

Rule 1: This first rule prevents anyone from accessing the FW-1 device directly, alerting the admins when there is such an attempt. No one should need to touch the FW-1 device aside from the admins.

Rule 2: The largest portion of the passing traffic, and the portion with the highest burst traffic levels, is the incoming VPN load for partners/distributors, etc. Since the application will be entirely web-based and deals with only one server, the FW1 can limit the connections to http/https directed at svc3app (the application webserver) and block all other traffic. In the same step, the rule will allow the management, internal, and ticket server networks in for the same requests.

Rule 3: The ticket server will be making SQL requests to the backend database server in the DMZ, so it will need these ports as well.

Rule 4: This fourth rule is for the syslog traffic passed to the syslogd server on the MGMT network. Many of the devices and several servers are logging all events to the internal syslog server(s), and the quantity of packets will be consistent enough to warrant a high placement in the rulebase in an effort to reduce processor load.

Rule 5: The primary E-Mail (Exchange/Groupware) server is on the corporate LAN. Since there is too great a risk in opening all necessary protocols to the VPN network, users will access the Outlook Web Access (OWA) Exchange feature to limit the need for a large quantity of ports.

Rule 6: Now the firewall can allow outgoing connections for the management and internal LANs. The Internet application list includes applications for Internet as well as for management of webservers.

Rule 7: The Exchange server needs mail from the mail-relay in the first service network to arrive at the Exchange server; this rule allows that to happen.

Rule 8: Another very minor risk is allowing use to run SSH from the VPN network to the MGMT network. Without this access, administrators would not have the ability to view the logs on the syslog server. From the MGMT network, the admin would be able to SSH to any machine that has a daemon and work as required, although without a GUI.
Note: It would be possible to set up X11 forwarding on the SSH server in the MGMT network to use for a GUI, however it would be limited in function when dealing with Windows systems.

Rule 9: There is likely to be quite a few broadcast-type protocols on the network, particularly NetBIOS (Windows), which could trigger the firewall's alerting in the last rule. This rule will drop any packet received on the broadcast address of any interface.

Rule 10: The last rule is the final level of protection. The list above has given permitted activity on all ports necessary for functionality, so anything else is unexpected and should never appear. If such traffic appears on the network, the firewall will let the admins know. Since there are users coming in on VPN, there is a slight chance this rule will get some false positives as those

users put in incorrect commands and sites; the VPN rules should prevent the majority and any that leak through would be the foundation for contacting Cisco for support on the issue.

C. Auditing

C.1. Concepts and Planning

With the complete Firewall/VPN configuration, it is possible to focus on the maintenance and testing of the configuration. There are a variety of tools available for this purpose; most are either identical or quite similar to the tools used in the initial scans performed by attackers. For the purposes of testing, this audit will rely on a pair of solid tools, Nessus and Nmap.

In determining the targets of the scan and which tool to use, the tools themselves must be considered. Nmap's strength lies in its ability to use a variety of packet types and sizes to best manipulate the target device. For this reason, Nmap v2.53 will primarily be used to locate weaknesses in the firewall configuration, with the side goal of ensuring proper protection and function of other devices. Nessus v1.0.9 is a better tool for finding weaknesses in applications and services; therefore it will be run on the devices in the various service networks. Additionally, Nessus will indicate if there are any issues with the server or firewall configuration that might prevent users from accessing the resources available on these networks.

In order to receive the most complete results, the audit will be run in several steps.
Note: For testing of devices not externally accessible, the scanning machine will be placed on the physical subnet with the hardware in question.

Step 1: Compile a list of expectations: What services are available? What servers should be accessible? How should the devices respond to the probes/scans? Build a plan to both test the strength of the design as well as testing the function of that design.

Step 2: Run a preliminary Nmap scan during off-hours on all devices. Document open ports and OS information for further testing and review. Use several machines with different Internet connections for complete results.

Step 3: Run a complete Nessus scan during off-hours on all devices. Ensure all proper updates are in place and ensure proper response on all services and ports. Again, use several with different Internet connections for complete results.

Step 4: Review results from primary scan. Track down and correct any detected issues and questionable output. All successive scans and verification will use this information to augment and direct scans looking for weaknesses.

Step 5: Run Nmap during normal business hours on all devices. Log all open ports, looking particularly at possible ports opened for transactions services and applications. As always, run from several different sites/connections for the most complete result. Try to time-sync as possible so information can be best correlated.

Step 6: Configure a packet sniffer, such as Ethereal, on a Linux machine in the DMZ to ensure no undesired traffic is escaping the firewalls, and to ensure the VPN and SSL data transactions are indeed encrypted.

Step 7: Through 3rd party, run ISS (Internet Security Scanner) on all external devices during non-business hours. Although this could miss some problems due to lack of load, the business needs outweigh the limited risk of a missed security issues. Compare results to output from other applications. A scan of this nature, run by a

3rd party consulting firm with assistance in review and comparison to existing information would likely run $2000-$4000 depending on the time required.

Step 8: Compare the results of the scans with the expectations from the first step and discuss plans for corrective action where necessary. Look over IDS to ensure detection of particular scans and check server logs for alerting messages that may indicate service disruption or failure.

The goal of this audit is primarily to ensure the devices installed would be capable of protecting the resources and services provided by the company without impacting the ability of those services from functioning correctly. A secondary goal is to ensure all installed services and applications are secure enough to resist an attack that would pass unfettered through the firewalls/VPN. A side effect will be the testing of the IDS as these probes (attacks) are picked up and logged.

Through complete logging and record keeping, the information gathered will be very useful in determining the capability of each piece of hardware to protect the resources within. Each scan can be compared to the next with the differences being pointed out and questioned. Since the some scans would be run simultaneously and others would be run at varied times of different days, the total quantity of information would be sufficient to ensure a comprehensive picture of the level of security in place.

C.2. Process and Results

C.2.1. Step 1: Expectations

The basic idea for this audit is to ensure the border protection and firewalls are functioning as intended. In order to know whether everything is working as would be expected, it is necessary to know what the requirements and expectations are. The following table describes the source of each scan and what would be the expected response. This audit requires taking the position of scanning all ports rather than focusing on specific ports unless there are particular rules or configurations that need direct testing.

There are several layers to this design, so the expectations will be extensive.

Nmap Source	Result
Internet/DMZ	11.5.5.1, no ports open 11.5.5.2, no ports open 11.5.5.3, TCP: 80, 443 11.5.5.4, TCP: 53, UDP:53 11.5.5.5, TCP: 25 11.5.5.6, TCP: 51, 500
Service1	172.16.0.1, no ports open 172.16.0.3, TCP: 21, 80, 443, 3389 172.16.0.4, TCP: 22, 53, UDP: 53 172.16.0.5, TCP: 22, 25

	172.16.0.253, TCP: 80, 443
Ticket1 (PIX)	172.16.1.1, no ports open 172.16.1.2, TCP: 80, 443, 3389
Vpn (PIX)	172.16.2.1, no ports open 172.16.2.2, no ports open
Private (PIX) Public (FW1)	172.16.3.1, no ports open 172.16.3.2, no ports open 172.17.3.10, UDP: 514 172.18.0.3, TCP: 25
Ticket (FW1)	172.17.2.1, no ports open 172.17.2.1, TCP: 80, 443, 3389 172.17.1.4, TCP: 1433, UDP: 1433
Vpn (FW1)	172.17.0.1, no ports open 172.17.0.2, TCP: 22, 443 172.17.1.2, TCP: 80, 443 172.17.3.10, UDP: 514
Svc3	172.17.1.1, no ports open 172.17.1.2, TCP: 80, 443, 3389 172.17.1.4, TCP: 1433, 3389 172.17.1.5, TCP: 22, 53, UDP: 53 172.17.3.10, UDP: 514

Running these tests will be a lengthy process. The objective is to run a scan on every physical network, performing a scan against the entire subnet looking for unusual or unexpected responses. The audit process uses several machines to improve the speed at which results are returned, however the time investment will be significant.

C.2.2. Step 2: Nmap v2.53 off-hours

The audit starts with an Nmap scan on each hardware device during non-business hours. Since it is off peak, these scans can be fairly aggressive with less regard for obstructing traffic and more emphasis on completeness. First is a plain TCP connect scan, without a preliminary ping to ensure the host is live. Many of the public Internet hosts are actually NAT'ed, so each scan will only reveal the ports that are open to the server, all others being marked as filtered.

```
[chris@pengie chris]$ sudo nmap -v -O -sT -P0 11.5.5.3

Starting nmap V. 2.53 by fyodor@insecure.org ( www.insecure.org/nmap/ )
Initiating TCP connect() scan against  (11.5.5.3)
Adding TCP port 80 (state open).
Adding TCP port 443 (state open).
The TCP connect scan took 744 seconds to scan 1523 ports.
Interesting ports on  (11.5.5.3):
(The 1521 ports scanned but not shown below are in state: filtered)
Port       State       Service
80/tcp     open        http
443/tcp    open        https

TCP Sequence Prediction: Class=truly random
```

```
                    Difficulty=9999999 (Good luck!)

Sequence numbers: 6C43548D 689D6BCD 3D22AD1A 54207D84 3FB5C4D3 3C8C96A0
Nmap run completed -- 1 IP address (1 host up) scanned in 759 seconds
```
Note: Edited for brevity.

The Linux workstation "pengie" has been placed in the DMZ to run a scan against the webserver public IP address. As expected, the only ports shown to be open are HTTP and HTTPS, even though the webserver is running Windows2000 and may have several other ports open. The PIX firewall does not respond any differently to the stealthier SYN scan, still showing only 2 open ports as above.

The "pengie" workstation is next placed on a broadband cable network and runs a scan to test if the ACL on the border router is functional. A second Linux box (11.5.5.253) comes into play here, with a telnet service configured to listen to specific ports blocked by the router, such as 23 and 31337, and then scan the IP of the Linux box for that port. Since the port may be filtered, "pengie" will run an ACK scan to try to probe the router to tell as much as possible about this port.

```
[chris@pengie chris]$ sudo nmap -v -O -sA -p 31337 -P0 11.5.5.253

Starting nmap V. 2.53 by fyodor@insecure.org ( www.insecure.org/nmap/ )
Initiating ACK scan against  (11.5.5.253)
The ACK scan took 36 seconds to scan 1 ports.
Interesting ports on  (11.5.5.253):
Port       State       Service
31337/tcp  filtered    Elite

Nmap run completed -- 1 IP address (1 host up) scanned in 222 seconds
```
Note: Edited for brevity.

This result indicates the border router is indeed blocking port 31337, so even though the Linux box has in.telnetd listening on port 31337 it is not possible to directly contact the machine on that port. This will repeat this process for each of the ports the border router has denied in an ACL.

C.2.3. Step 3: Nessus v1.0.9 off-hours

Once the Nmap scans have been completed, Nessus will be run to shed more light on the ports Nmap has found to be open. Nessus is a tool designed to locate and pinpoint weaknesses in applications or services. It produces clear reports that can be easily used as checklists for updating and patching weaknesses or loopholes.

Nessus will be used to paint a good picture of the whole network security design by placing servers on a variety of network connections, from each physical Ethernet subnet to several broadband connections from varied providers. Each connection will give us some information on the services available and the weaknesses those servers have. *In the interest of brevity, a Nessus scan against the external webserver can be found in Appendix D.*

C.2.3. Step 4: Review Results and Patch

With the information from 2 scans in hand, it is now possible to compare the results to what was expected. This comparison is done it ensure resources are being made available as was originally intended, and that none of these resources have been compromised.

Perhaps when the scan was run against the DNS server something was discovered outside the expectations, such as an additional open port:

When the scanner was on the Internet, this was discovered:

```
Starting nmap V. 2.53 by fyodor@insecure.org ( www.insecure.org/nmap/ )
Initiating TCP connect() scan against  (11.5.5.4)
The TCP connect scan took 813 seconds to scan 1523 ports.
Interesting ports on  (11.5.5.4):
Port       State       Service
53/tcp     open        domain
80/tcp     open        http
```

Note: Edited for brevity,

However, when on the service network, the scanner received this:

```
Starting nmap V. 2.53 by fyodor@insecure.org ( www.insecure.org/nmap/ )
Initiating TCP connect() scan against  (172.16.0.4)
The TCP connect scan took 1 seconds to scan 1523 ports.
Interesting ports on  (172.16.0.4):
(The 1521 ports scanned but not shown below are in state: closed)
Port       State       Service
22/tcp     open        ssh
53/tcp     open        domain
```

Note: Edited for brevity,

It would appear in this example that the firewall is somehow improperly configured. Although the DNS server is certainly not running any HTTP service as is expected, the firewall is accepting and forwarding a connection to the DNS server without regard for that fact. A review of the firewall rules may find an improper static/conduit statement pair, perhaps a typo that has also affected the primary DNS or simply an additional unneeded command.

The Nessus scan provides more information on the services that are accessible with regards to the existence of known weaknesses or poor configuration. Since the servers are running Microsoft-based application, tools like this are very useful in ensuring a solid, secure configuration. Perhaps Nessus returned an issue when it scanned the webserver in the first service network from inside the first network.

```
. Vulnerability found on port netbios-ssn (139/tcp) :

   . It was possible to log into the remote host using a NULL session.
   The concept of a NULL session is to provide a null username and
   a null password, which grants the user the 'guest' access.


   . All the smb tests will be done as
    '/''
```

If this issue were to be visible on the outside, the machine would be only protected by the border router configuration that prevents incoming TCP connections. Even in the DMZ, it is a serious risk as it leaves the server open to SMB connects, useful to someone who has control of one machine in the service network and wants to expand his or her control. NetBIOS is a serious weakness, and should be dealt with quickly and effectively.

C.2.3. Step 5: Run Nmap during Business Hours

The next step is to run the Nmap tool once again on the same devices, however this will be done during business hours. Scans have been run during off-peak hours providing more insight into how the hardware and software is unlikely to fail under the scan, however the aggressiveness of the scan is reduced to prevent inadvertently DoS-ing these resources. This scan should return the same information as the night scan, any differences will be looked at very hard to ensure there are no local applications which might be either interfering with the firewall(s) or completely overriding them.

C.2.3. Step 6: Sniff Local Networks

There is a fair collection of data showing us the strengths and weaknesses of the security architecture at this point. It is possible to start to lay out a design of what is properly defended and what is not by using the results from previous scans. Now it is necessary to ensure no internal network traffic is being sent to the public Internet. A Linux box running the application Ethereal will be placed on the DMZ an in the public network between the firewalls to make 5-minute long captures at different business hours. This Ethereal trace will be checked for private-IP sourced traffic and for unencrypted data passing to and from encryption devices or servers to verify all traffic is being handled as expected.

C.2.3. Step 7: Get Outside Perspective

It is time to get the view of a third party that will objectively look at the information so far gathered, and use that data to design and implement a test using the ISS tool. With the use of an application not included in the previous scans (ISS), it is possible that additional issues to be corrected will be discovered. The ISS output will also be used to verify the information gathered by Nmap and Nessus to add credence to the data.

C.2.3. Step 8: Review Logs, Results and Patch

Having completed the scanning and begun preparations for a final review, it is time to look at the infrastructure to see what information each server was able to gather. The primary concerns are applications with a known record of being either unreliable or easily hacked. IIS and DNS servers will be reviewed for any error logs regarding the health and stability of the respective services. A review of the IDS logs will verify they have captured the scans and dealt with the information appropriately.

By this point there is large quantity of data that has been reviewed to ensure a proper network structure and attack responses. The perspective brought to the table by the third party company helps to mould the results from the previous steps into a plan to correct any discovered issues and keep the focus on the weaknesses for further or continuous review.

C.3. Conclusion

Security is not a single process that, once complete, can be set aside for other tasks. It is a continually developing and evolving art that takes time and energy to keep up to date. This audit may have shown the security level to be high and found very few weaknesses to be concerned about, but tomorrow's new perspective on the TCP/IP standard might result in a potential weakness that was unexpected and unplanned for – ala 'Ping of Doom'. This audit process needs to be repeated and revised continuously, and planned top to bottom audits with external support should be scheduled on a regular basis to maintain the level of security found in this single audit.

Not to be forgotten are the critical parts of authentication used every day: passwords. While this audit is not specifically looking at passwords, even if every device was locked down as tightly as possible, a simple password could make it all for naught. Routine usage of password cracking applications, 'Crack' for Unix machines and 'L0phtCrack' for Windows-based machines, can help reduce the risk of leaving the door open and the lights on for would-be attackers. Auditing is a part of life in the world of security, without it is not possible to be certain of what happens inside a public network.

D. Attacking a Network

Note: This section uses the design found in Keith Wilcox' SANS Parliament Hill GCFW Practical Assignment. The original submission can be found at http://www.sans.org/y2k/practical/Keith_Wilcox.doc.

D.1. Attacking the Firewall Directly

The first step in any attempt to gain access to a network or server is research. There are a variety of information resources available on the Internet with documentation on weaknesses and exploits for any one of the popular, some less popular, firewall applications, servers and services. When laying plans for such an attempt, the question is less one of preparedness and more one of timing: Find the weakness before the defending site does and exploit it.

There is little leniency in time for this particular attack, which makes a frontal assault the only alternative. Given this fact, a target must be chosen based on the number of known weaknesses and its ability to detect attacks. For this reason, the selection is be a CheckPoint FW-1-protected network.

FW-1 is a good choice for this circumstance for the same reasons as it wasn't placed as the primary firewall in the previously described network design: It's logging is fairly limited, there are several known weaknesses, and the occasional unexpected feature pops up which might still not be patched. The end hope is that the attacks being made can be disguised or obfuscated enough to ensure the longest possible timeframe to run the attacks.

The SecurityFocus website has a list of weaknesses related to FireWall-1 and a few have known exploits and tests. As would be hoped for in any firewall, the list is rather short and does not include many useful attack options. The most interesting option comes from another website, http://www.dataprotect.com/bh2000, where there is both a lengthy discussion on the weaknesses of FW-1 in certain versions as well as the exploit code from which and attack can be based.

In order to use this method some information gathering will be required. There is one good piece of information: the documentation to the original FW and network configuration. This document indicates there is a preemptive measure taken in the firewall ruleset to deny any packet destined for the firewall with an alert sent to an admin. While this will be unlikely to affect the attack itself, this information will be used to the attack's advantage. A simple Linux box will be configured to send spoofed ICMP, TCP, and UDP packets directly to random ports the firewall at random intervals 24/7 for several weeks to set off the alert as often as possible regardless of time of day (or night).

In order for the attack to succeed, the IP address of the management workstation(s) will be required. It is possible this information could be gained by running the 'fwnone' application found on the dataprotect website. The script repeatedly executes, incrementing the management workstation IP guess through the private IP range, followed by public IP ranges, until it manages a hit. If the application receives 'Deny' messages, there is a fair chance the firewall has been patched against this particular weakness.

Assuming the IP address of the management workstation can be found using this method, or if the network design document is accurate, the attack will move to cracking an authentication key using the 'fw1bf' application, once again from the dataprotect site. If a valid key is returned, the 'fw1skey' tool will be used to unload the filter module and open the firewall on all ports and addresses.

In a more realistic sense, there is little likelihood this weakness has not been patched. Attacking a firewall is mostly a question of timing, as mentioned before. If a new weakness is found before the firewall admins can locate a patch it would be possible to use the information to directly attack the firewall. Firewalls are not necessarily a "silver bullet", however they do deserve some level of respect in the security arena until a good piece of information comes around.

D.2. DoS/DDoS

Attacking the firewall is frustrating work, so the next task is to perform a DDoS attack from a legion of 50 or so compromised PCs with broadband Internet access. DDoS is a simple way for the beginner to make their mark on a company or service. As the site GRC DDoS Diary shows, it doesn't take much knowledge to cause significant destruction, and the ability to turn back such attacks can be rather limited.

Once again the target is running CheckPoint FW-1, a stateful firewall. Perhaps the easiest method to use is an attempt to fill the state cache. There are several DDoS possibilities in stateful firewalling from the fact that each connection (incoming and outgoing) must be cached in order to properly handle packets for those connections. Additionally, any packet stream that has been broken up due to MTUs would have to be held for reassembly by the firewall before being handed to the destination.

The attack comes in the form of a TCP SYN attack against the target webserver. A series of hosts with broadband Internet access are sending TCP SYN packets to the webserver, with the effect of filling the external state cache on the firewall and drowning legitimate traffic. Some of the source hosts are taking the added step of spoofing random source addresses to add to the load. At some point the state table will reach critical mass, and firewall services will collapse under the table's weight. As a side effect, even as the firewall's functionality slowly degrades, bandwidth will be severely restricted and access to public WWW services will be degraded at best. CheckPoint FireWall-1 does have an application built in to handle such floods, however most documentation indicates the capability of this application seems to be limited.

'(Distributed) Denial of Service' attacks are already difficult to prevent. The Internet stands at a turning point as Windows9X, the most popular home desktop OS, is limited in its ability to act as a DDoS client due mainly to the fact that the TCP sockets function is not completely compatible with the TCP standard. As a result, Win9X PCs have the limitation of being unable to spoof their source IP addresses, something Windows2000 and the upcoming WindowsXP will be able to accomplish. Fortunately, the numbers of true Unix-compatible sockets is currently limited, lending slight good news in the temporary cessation of DDoS attacks: router ACLs.

GIAC Enterprises will have a good working relationship with its upstream ISP as a result of the quantity of business it does over the ISP's data networks. Using this relationship during the DDoS could be very useful, particularly if the ISP will make upstream ACL changes based on the source addresses indicated in the attacks. Some will be spoofed; however with some careful logging the spoofed addresses will become more apparent as they appear in one attack and not in the next, while the more normal addresses will be consistent.

Blocking these attacks upstream from local resources will remove the load off the data services as well as limit the effect on system time the attacks are causing. The potential downside would be the clever attacker who has sniffed daily local IP traffic could spoof the addresses of more common customers and suppliers. For these, difficult decisions would need to be made based on the needs of the company, the level and type of access a particular customer would need, and the alternatives available.

D.3. Attacking A Protected Host

An alternative to attacking a firewall would be to completely bypass the firewall altogether and aim at an application server. It's pretty well accepted that even with known weaknesses, firewalls are very difficult to attack. An application running on a protected server would be a much easier target – as the Code-Red worm has recently shown with it's escapades with Microsoft's IIS.

The first step in a successful attack is, once again, planning. Hiding the attack is a primary concern, since most locations would have some sort of intrusion detection looking for the very activity to be performed. Launching a mild DDoS against the site as the attack is launched would be one method to limit the quality of information available, particularly if the DDoS could include packet types similar to those used in the attack itself.

Information gathering on the network and server to assaults is also needed. The site chosen for the attack is running Windows-based Internet servers, including the IIS services for WWW and FTP. There is a decent selection of working examples of the weaknesses in an improperly monitored and maintained IIS server, and with these exploits will attempt to take one over. The Nessus tool will be used to find any known unpatched weaknesses and to look for any incorrect configurations. Nmap can also be used to generate traffic and to probe open ports to ensure the server is running the expected OS. During any scan, a light DDoS will be run to cover the tracks of the scan.

Using the information gathered by these scans, it is possible to select the weakness to be exploited. Another scan is now run to augment the already gathered information; a scan done by an application called Whisker. Whisker is a CGI script scanner designed to find weaknesses in websites and the code behind them. As this last scan is run, several security websites will be checked for documentation and information on potential weaknesses and exploits.

This attack occurs in the lull after a storm; the level of alertness amongst most administrators is at an all-time high due to the Code-Red and Code-Red 2 IIS worms. Any machines that are still

susceptible to the known exploits are either already rooted or are not of any value – unused, old and/or slow.

E. Conclusion

Regardless of the network design, the amount and capability of the hardware, or the level of monitoring in place, the only thing good perimeter security and firewall configurations can provide is time.

The question is how much time; with a design with multiple layers using different hardware and software solutions time can be maximized. Given a skilled, determined attacker, each of these layers will simply act as an opaque glass wall – just enough resistance and just enough risk of injury to force the attacker to slow down and plan the next move.

In the time these attacks are made, the network security team will have the opportunity to cut off the attack, determine the weakness utilized, and make the necessary changes to prevent such an incursion again. However with employees coming and going, knowledge of the security systems will become more commonly known – and the attackers will have more and more to work with.

Only with constant update, monitoring, and threat analysis, along with the occasional tweak or reconfiguration, can this section of the network infrastructure continue to do it's assigned job: Protect the valuable electronic resources within the company while making that same information available to the world.

Appendix A. Complete Border Router ACL & Pertinent Configuration
Note: Active ports, IPs, and increasing port activity determined on 7/26/01

```
no ip unreachables
no ip direct-broadcast
no service tcp-small-servers
no service udp-small-servers
no service finger
no ip http server
no ip bootp server
no snmp
no ip source-route

access-list 101 deny ip 192.168.0.0 0.0.255.255
access-list 101 deny ip 172.16.0.0 0.0.255.255
access-list 101 deny ip 10.0.0.0 0.0.0.255
access-list 101 deny ip 12.0.0.0 0.255.255.255
access-list 101 permit tcp any 11.5.5.3 eq 80
access-list 101 permit tcp any 11.5.5.3 eq 443
access-list 101 permit tcp any 11.5.5.5 eq 25
access-list 101 deny tcp any any eq 111
access-list 101 deny tcp any any eq 4329
access-list 101 deny tcp any any eq 23
access-list 101 deny tcp any any eq 21
access-list 101 deny tcp any any eq 6346
access-list 101 deny tcp any any eq 515
access-list 101 deny tcp any any eq 1141
access-list 101 deny tcp any any eq 2049
access-list 101 deny tcp any any eq 1999
access-list 101 deny tcp any any eq 8000
access-list 101 deny tcp any any eq 2301
access-list 101 deny tcp any any eq 27374
access-list 101 deny tcp any any eq 27015
access-list 101 deny tcp 64.166.1.197 any
access-list 101 deny tcp 153.91.109.254 any
access-list 101 deny tcp 62.254.128.5 any
access-list 101 deny tcp 24.27.84.166 any
access-list 101 deny tcp 211.175.33.40 any
access-list 101 deny tcp 211.185.195.1 any
access-list 101 deny tcp 198.4.241.2 any
access-list 101 deny tcp 217.84.90.33 any
access-list 101 deny tcp 210.164.255.73 any
access-list 101 deny tcp 209.87.111.20 any
access-list 101 deny tcp any any eq 31337
access-list 101 deny tcp any any eq 12345
access-list 101 deny tcp any any eq 12346
```

```
access-list 101 deny tcp any any eq 137
access-list 101 deny udp any any eq 137
access-list 101 deny tcp any any eq 138
access-list 101 deny udp any any eq 138
access-list 101 deny tcp any any eq 139
access-list 101 deny udp any any eq 139
access-list 101 deny tcp any any eq 49
access-list 101 deny udp any any eq 49
access-list 101 deny tcp any any eq 130
access-list 101 deny tcp any any eq 131
access-list 101 deny tcp any any eq 132
access-list 101 deny tcp any any eq 1467
access-list 101 deny tcp any any eq 1741
access-list 101 permit udp any any
access-list 101 permit tcp any any
```

These rules were applied to all traffic coming in on serial0.

```
interface serial0
ip access-group 101 in
```

```
access-list 102 deny 10.0.0.0 any 0.255.255.255
access-list 102 deny 172.16.0.0 any 0.31.255.255
access-list 102 deny 192.168.0.0 any 0.0.255.255
access-list 102 deny 127.0.0.0 any 0.255.255.255
```

These rules were applied to all traffic coming in on ethernet0.

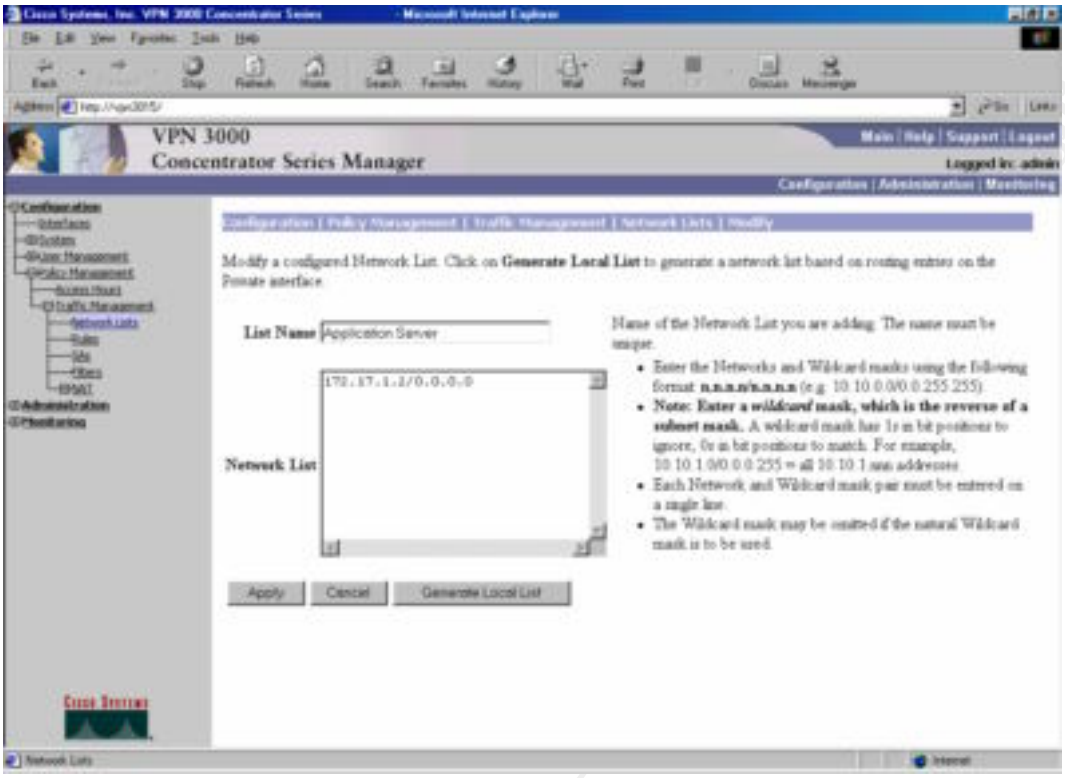
```
interface ethernet0
ip access-group 102 in
```

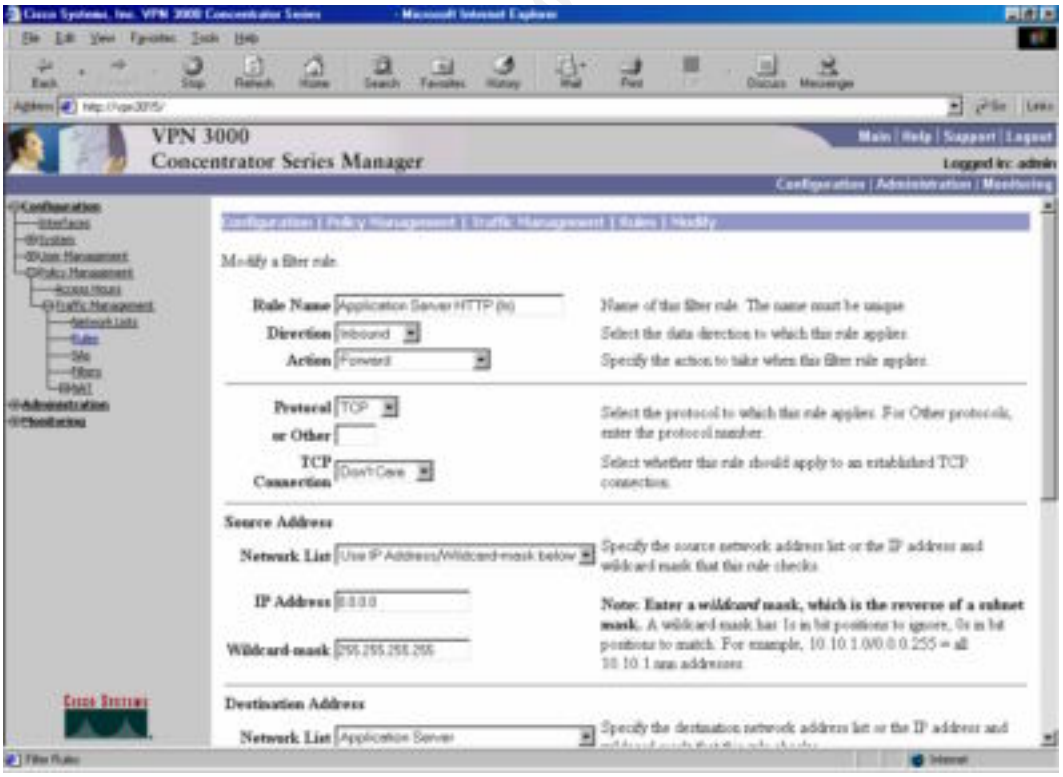
Appendix B. Complete PIX Firewall Security Configuration
Note: This is based on v4.4 documentation – there may be minor inconsistencies with newer releases.

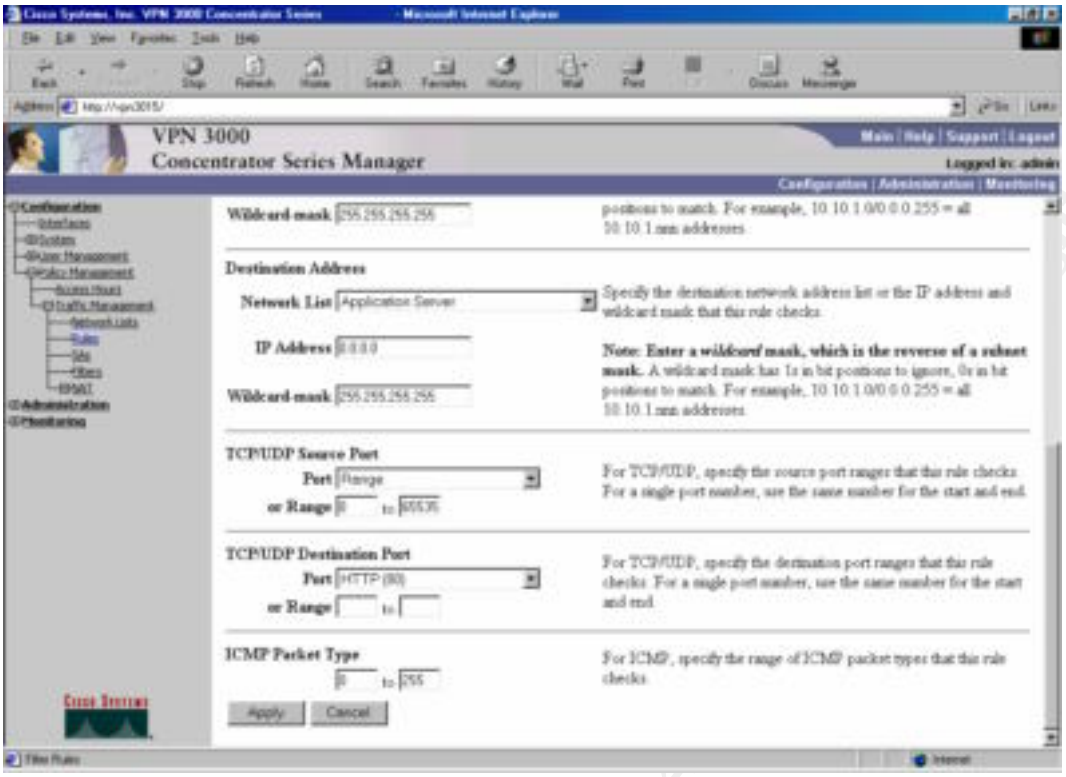
```
nameif ethernet0 public security0
nameif ethernet1 service1 security10
nameif ethernet2 ticket1 security15
nameif ethernet3 vpn security20
nameif ethernet4 private1 security25
nameif ethernet5 service2 security5
ip address public 11.5.5.2 255.255.255.0
ip address service1 172.16.0.1 255.255.255.0
ip address ticket1 172.16.1.1 255.255.255.0
ip address vpn 172.16.2.1 255.255.255.0
ip address private1 172.17.0.1 255.255.255.0
ip address service2 172.16.3.1 255.255.255.0
nat (private1) 25 172.17.0.2 255.255.255.255
global (public) 25 11.5.5.254 netmask 255.255.255.0
global (service1) 25 172.16.0.254 netmask 255.255.255.0
static (service1,outside) 11.5.5.3 172.16.0.3 netmask
255.255.255.255
conduit permit tcp host 11.5.5.3 eq www any
conduit permit tcp host 11.5.5.3 eq 443 any
static (service1,outside) 11.5.5.4 172.16.0.4 netmask
255.255.255.255
conduit permit tcp host 11.5.5.4 eq domain any
conduit permit udp host 11.5.5.4 eq domain any
static (service1,outside) 11.5.5.5 172.16.0.5 netmask
255.255.255.255
conduit permit tcp host 11.5.5.5 eq smtp any
static (vpn,outside) 11.5.5.6 172.16.2.6 netmask 255.255.255.255
conduit permit tcp host 11.5.5.6 eq isakmp any
conduit permit tcp host 11.5.5.6 eq esp any
static (ticket1,service1) 172.16.0.253 172.16.1.2 netmask
255.255.255.255
conduit permit tcp host 172.16.0.253 eq www any
logging host private 172.17.3.10
logging trap warnings
logging timestamp
logging on
outbound 10 deny 172.18.0.0 255.255.0.0 any any
outbound 10 permit 172.18.0.0 255.255.0.0 smtp tcp
apply (service1) 10 outgoing_src
outbound 20 deny 172.16.0.0 255.224.0.0 any any
outbound 20 permit 172.18.0.0 http tcp
outbound 20 permit 172.18.0.0 443 tcp
outbound 20 permit 172.18.0.0 ftp tcp
```

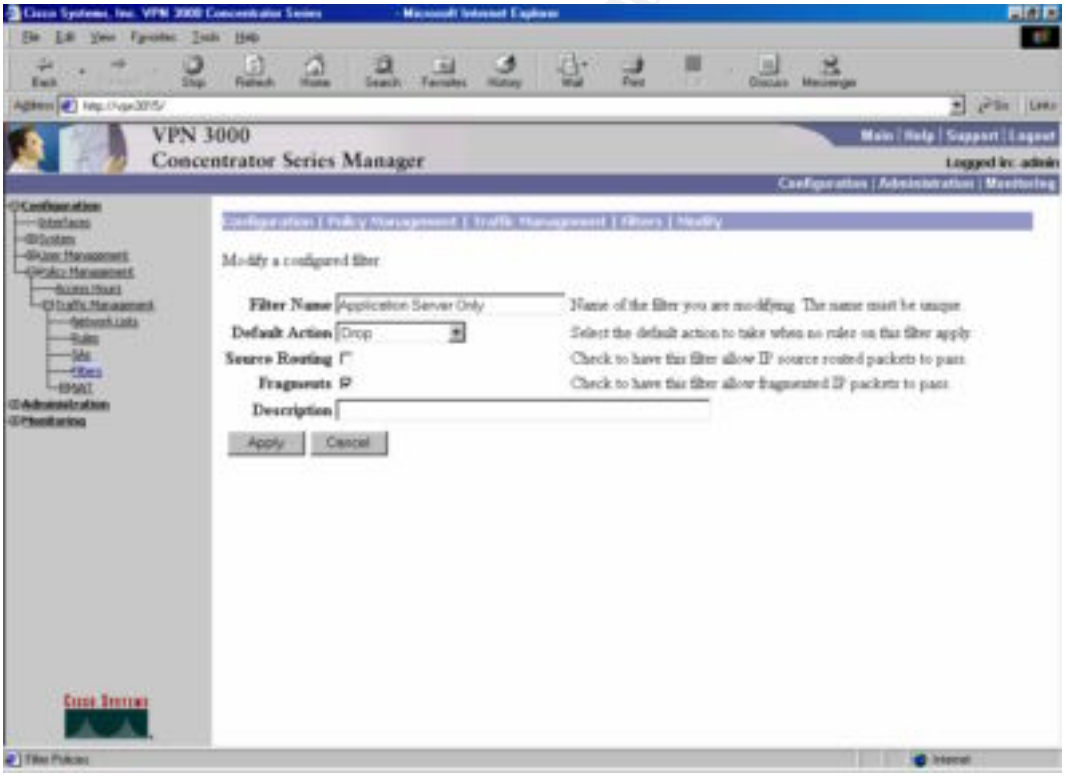
```
outbound 20 permit 172.18.0.0 ssh tcp
outbound 20 permit 172.18.0.0 53 tcp
outbound 20 permit 172.18.0.0 53 udp
apply (public) 20 outgoing_src
```

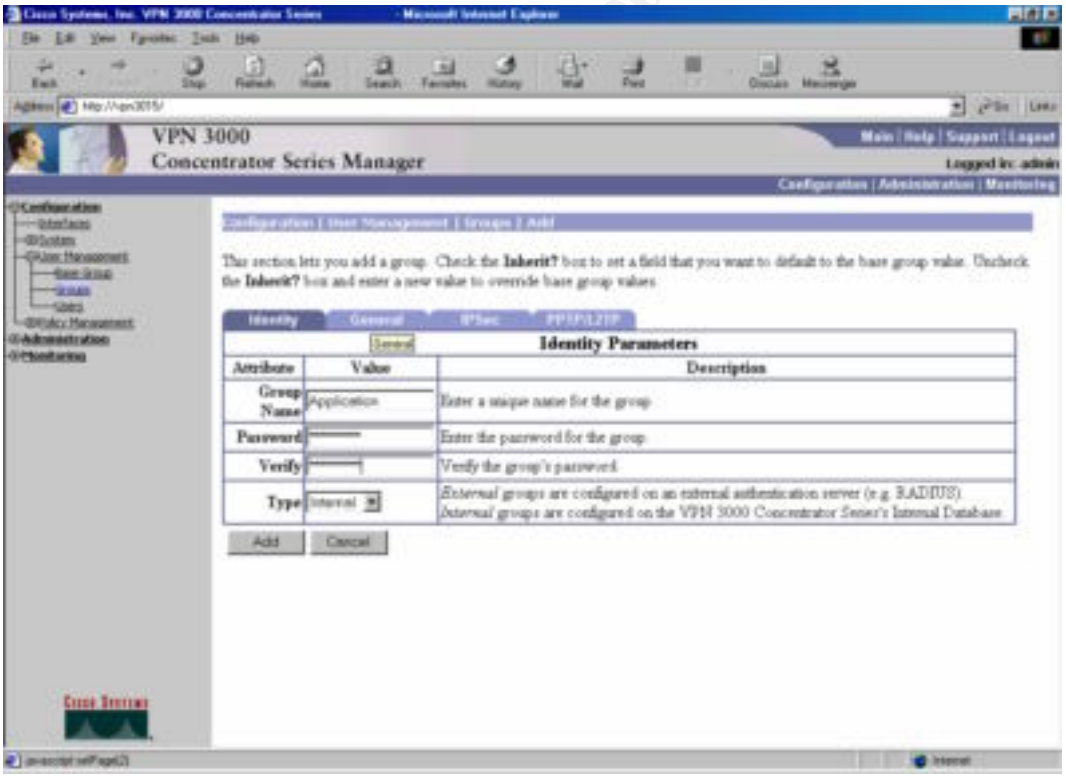
Appendix C. VPN Configuration Output

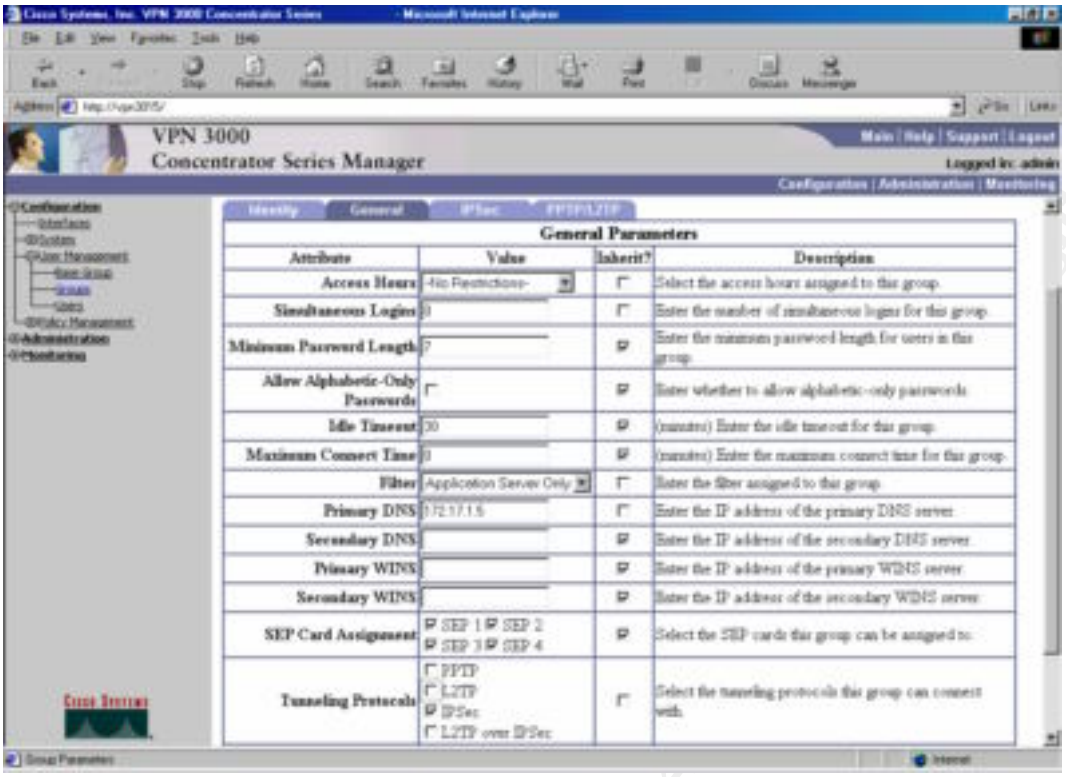


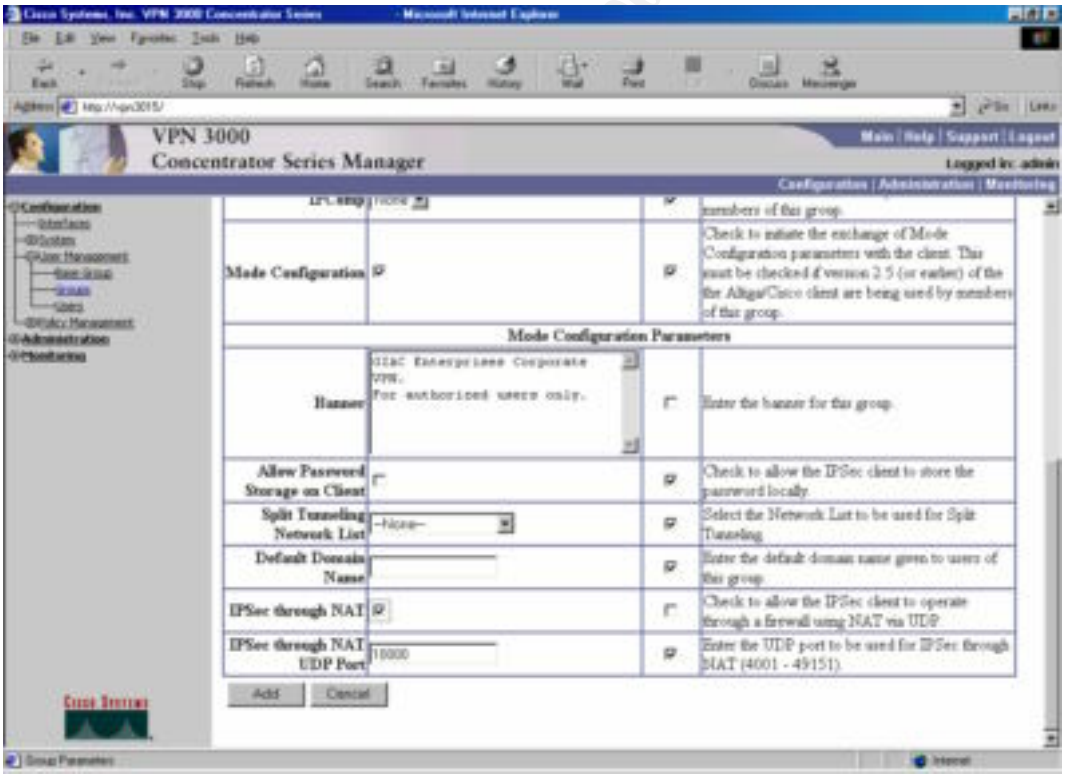












Appendix D. Nessus Sample Output

```
Nessus Scan Report
------------------

SUMMARY

 - Number of hosts which were alive during the test : 1
 - Number of security holes found : 0
 - Number of security warnings found : 1
 - Number of security notes found : 2

TESTED HOSTS

 172.17.1.2 (Security warnings found)


DETAILS

+ 172.17.1.2 :
 . List of open ports :
   o general/udp (Security notes found)
   o http (80/tcp) (Security notes found)
   o general/tcp (Security warnings found)

 . Information found on port general/udp

    For your information, here is the traceroute to 172.17.1.2 :
    172.17.1.2

 . Information found on port http (80/tcp)

    The remote web server type is :
    One Fast Webserver OFWv1.4

    We recommend that you configure your web server to return
    bogus versions, so that it makes the cracker job more difficult

 . Warning found on port general/tcp

    The remote host uses non-random IP IDs, that is, it is
    possible to predict the next value of the ip_id field of
    the ip packets sent by this host.

    An attacker may use this feature to determine if the remote
    host sent a packet in reply to another request. This may be
    used for portscanning and other things.

    Solution : Contact your vendor for a patch
    Risk factor :
     Low


-------------------------------------------------------
This file was generated by the Nessus Security Scanner
```

Appendix E. Basic Secure System Configuration

A. RedHat Linux (DNS example)

The RedHat server installation is performed using the vendor-supplied media. This will automatically install several necessary services and applications, several unwanted services, as well as the option to install further applications.

This machine will be configured to be as secure as possible and each part of the install must be considered carefully. There have been some troubles in the past with logs overflowing during an attack causing buffer overflows in syslog and other services if not properly configured. To add a layer of protection several partitions will be created and mounted as separate volumes: /var, /chroot, /etc, /.

Next is the selection is the services and packages that are required. The list is short, and could be shorter depending on what will be done manually through source code and packages. The stock services syslogd, openssl, and openssh will be used, albeit upgraded before the server is placed into production. The BIND services will be compiled from source code later in this process.

User accounts are created, limited to a small quantity of users who will be required to use strong passwords and frequent (30 days or less) password changes. SSH will be used so most of the user logins will be through SSHv2 with public-private keys. This will make passwords slightly less of an issue for accepted users.

All unnecessary services are removed from startup after the first boot. RedHat includes an application called 'ntsysv' which is an easy method to change the bootup services. The list of applications required by this server is short: crond, keytable, network, random, rawdevices, sshd, syslog. The remainder can be removed from startup, and removed completely if it is certain they will be unneeded.

Every remaining service will be updated to the most current vendor-supplied versions by visit http://www.redhat.com and downloading the necessary patches and fixes.

BIND9 will be used to handle all the nameservice needs. BIND is available for download in source code form from ISC's website at http://www.isc.org; the most current release version, 9.1.3 as of this writing, will be downloaded. There are a few configuration options that depend on hardware configuration; otherwise a simple configure/make is all that is required.

BIND will be configured to run setuid and in a chroot jail. There is a good chroot/setuid how-to on http://linuxdocs.org that describes the steps to build the chroot jail and how to configure BIND to work properly inside that jail. The purpose of a chroot/setuid jail is to limit the damage a hacker could do should they manage to overrun the service; in a normal environment BIND runs as a root equivalent account leaving the system open if someone can exploit the BIND application. In this environment, that person would be stuck with the rights of a normal user in a small directory subtree containing only the files and information required by the BIND service.

There is no reason to use the 'root' account repeatedly to perform routing maintenance. Usage of the 'sudo' command will limit the need for administrative users to become root using the 'su' command. There are several commands admins may need access to and each user may have different levels of access, rights, and tasks. An example of a required application may be 'rndc', the application to start and stop the named/BIND service:

```
dnsadmin dnsserver = PASSWD: /usr/local/sbin/rndc
```

The final configuration step is preparing syslog. Adding a single line to the /etc/syslog.conf on this machine will tell syslog to report both to the local logging server as well as to the main syslog server. One line needs to be added to the configuration file:

```
*.* @syslog.ip.addr.ess
```

This is by no means a complete guide; the server will need additional file access rights changes, application and service configuration and the like. Once done, it will be ready for testing and deployment.

B. Windows 2000 (IIS example)

Windows 2000 has been selected for the web services, due mostly to the relative ease to which it can be used for most web applications. For that strength it does require significant planning and implementation time. It is easy for the developer to play with, but when it comes to security admins are fighting a constant uphill battle.

The Systems and Network Attack Center (SNAC), part of the National Security Agency (NSA), released several valuable reads on securing Windows 2000. These are highly recommend documents that anyone configuring a Windows 2000 server should read and keep on hand.. Unfortunately the NSA has taken the Internet site down making it somewhat more difficult to find the original documents.

A basic installation is the first step. During the installation, the server is created using 2 partitions: one for the Windows 2000 system and support files, and one for application files, both partitions using NTFS. The machine will be stand-alone, not a member of a domain or directory tree. All user and group accounts will be stored locally, and would need to be manually replicated to any server that it must directly access.

The installation will be customized as much as possible, selecting only the services and applications necessary. A few particular required services are IIS and Terminal Services, if admins will use them in lieu of console access. Any other applications will be installed and configured, under admin supervision, by the application provider at a later in this process.

Once the basic installation is complete, all known patches, services packs and fixes for the relevant services are installed. It will be necessary to continually check various system security sites for new hacks and weaknesses; IIS is under continuous scrutiny. IIS may install sample code; this must also be removed from the system.

Next is the creation groups and users. A good practice is the creation of a group for each administrative purpose, granting access to each user by assigning them to the appropriate groups. There will be several user types: software installers who will require full administrative rights, web administrators with access to the web folders on the second partition, as well as security admins who will need access to event logs and application service logs. Users will be required to use strong passwords, with changes required every 30 days.

Every service running on a server is a potential weak point, removing them is an important step. Page 4 of the "Guide to the Secure Configuration and Administration of Microsoft Internet Information Services 5.0" (later referring to as "NSA IIS Guide") goes into detail on what services would be required. This server is only going to handle HTTP and HTTPS requests, the FTP and SMTP services are also disabled.

IIS is the basis for this machine, it is best to spend some time looking over the requirements for the application to best lock down the server itself. The SNAC guide on securing IIS gives several good pointers and spends more time than available here going through the nuances of a secure IIS configuration.

A good practice is place the Internet directory structure on a separate drive from the system, which was already created during the install process. It will be necessary to change all the directory configurations when this is done to ensure all services use the new location. The new locations also should be locked down with access rights to limit access based on the previously mentioned groups, and make sure the Internet user has access according to the requirements of any installed applications.

The basics of locking down a large, inclusive operating system such as Windows 2000 is much more in-depth than the work done to a Linux server, due to the fact that the Linux server is much more modular and requires fewer services to run in a given environment. Spending a good deal of time working on the Windows 2000 server will reward in security and reliability, and with the company's business requiring these servers proper configuration will pay dividends in increased productivity and value.

Appendix F. Information Resources and References

Web-Based Information Sources

"Snort – The Open Source Network IDS", URL:
http://www.snort.org (19 Aug 2001).

Atkins, Todd, "SWATCH: The Simple WATCHer" 22 May 2001. URL:
http://oit.ucsb.edu/~eta/swatch (19 Aug 2001).

"Cisco Connection Online by Cisco Systems, Inc.", URL:
http://www.cisco.com (19 Aug 2001).

"Red Hat – Linux, Embedded Linux and Open Source Solutions", URL:
http://www.redhat.com (19 Aug 2001).

"Check Point Software Technologies. We Secure the Internet.", URL:
http://www.checkpoint.com (19 Aug 2001).

"Secure Network Solutions", Nokia on the Web. URL:
http://www.nokia.com (19 Aug 2001).

The SANS Institute, "Welcome to incidents.org – By The SANS Institute", URL:
http://www.incidents.org (19 Aug 2001).

"SecurityFocus", URL:
http://www.securityfocus.com (19 Aug 2001).

"CERT Coordination Center", URL:
http://www.cert.org (19 Aug 2001).

"SANS Institute: Information Security Reading Room" The SANS Institute – Home Page, URL:
http://www.sans.org/infosecFAQ/index.htm (19 Aug 2001).

Welch-Abernathy, Dameon D., "PhoneBoy's FireWall-1 FAQ", URL:
http://www.phoneboy.com (19 Aug 2001).

Muffet, Alec, "Alec Muffet's Home Page"URL:
http://www.users.dircon.co.uk/~crypto/index.html (19 Aug 2001).

"@stake LC3", @stake, Inc. URL:
http://www.atstake.com/research/lc3/index.html (19 Aug 2001).

Deraison, Renaud, "Nessus", 13 Aug 2001. URL:
http://www.nessus.org (19 Aug 2001).

Fyodor, "Insecure.org – Nmap Free Stealth Network Port Scanner,Linux/Windows/UNIX/Solaris Tools & Hacking" URL: http://www.insecure.org (19 Aug 2001).

Lopatic, Thomas, John McDonald, Dug Song, "A Stateful Inspection of FireWall-1" TueV data protect – Herzlich willkommen… 9 Aug 2001. URL: http://www.dataprotect.com/bh2000 (19 Aug 2001).

Gibson, Steve, "The Attacks on GRC.COM" Home of Gibson Research Corporation. 4 Jul 2001. URL: http://grc.com/dos/grcdos.htm (19 Aug 2001).

RainForestPuppy, "R a i n F o r e s t P u p p y.", W i r e t r i p URL: http://www.wiretrip.net/rfp/index.asp (19 Aug 2001).

"Linux Documentation Project", URL: http://ww.linuxdoc.org (19 Aug 2001).

Non-Electronic Information Sources

The SANS Institute, Track 2: Firewalls, Perimeter Protection and Virtual Private Networks, Volumes 2.1, 2.2, 2.3, 2.4, Presented May 2001.

Cisco, Configuration Guide for the PIX Firewall Version 4.4, June 99

Bales, Marion, William Stearns. Setting up automatic alerting in your Unix environment: January 26, 2001

McGovern, Owen R., Julie M. Haney, Guide to Securing Microsoft Windows 2000® File and Disk Resources, Version 1.0, April 19, 2001

Walker, William E., IV, Guide to the Secure Configuration and Administrator of Microsoft Internet Information Services 5.0, Version 1.0, October 11, 2000