GCFW

GIAC CERTIFIED FIREWALL ANALYST

Darren Page

Practical Assignment Version 1.9

Table of Contents

1. Abstract

This paper describes the Security Architecture and Firewall Policy for an imaginary company named GIAC Enterprises that deals in the sale of fortune cookie sayings.

The security architecture is detailed, a tutorial is provided on the external firewall configuration and the firewall policy of GIAC Enterprises is audited and analysed within this paper. Finally, three attacks on a previous GCFW practical assignment are proposed and discussed.

1.1 Document Overview

This document sets out in five basic areas:

- Security Architecture
- Security Policy and tutorial
- Firewall Policy Verification
- Design under Fire
- Appendix of References and Configurations

2. Security Architecture

GIAC Enterprises is an e-commerce company which deals in the online sale of fortune cookie sayings. GIAC is located in Sydney, Australia, but has a global customer base. Its suppliers are based in the US and Australia.
GIAC has approximately two hundred employees in Australia, but offers a flexible working environment that allows staff to work from home.

GIAC has recently established new partner relationships to provide language translations for new markets in Asia and Europe, which has additionally resulted in increased revenues from banner advertising with local language translations. GIAC has recently set up several sales offices in South-East Asia and Europe. These new sales offices will add an extra fifty staff.

Due to the recent expansion into new markets and an increasing global customer base, GIAC has upgraded its infrastructure to provide a highly available 7 x 24 e-commerce site. Revenue forecasts for the coming year are in excess of $100 million. This has more than justified the expense of the new security and e-commerce infrastructure.

GIAC has worked very hard to build a sound business case for their expansion and has gained financial backing from a number of large institutional investors to fund the new infrastructure. This is built around a multi-tiered application architecture with clearly defined presentation, application / business logic and data layers.

This architecture employs a dual layered firewall model. Additionally, no user or session state is maintained within the presentation layer; this is all managed by the application and data base servers.

2.1 Customers

Customers access the GIAC fortune cookies via the GIAC web site. The web site provides information on GIAC enterprises and details how to purchase online fortune cookies. Customers are required to log in with a valid username and password before being able to purchase any cookies and can obtain details of all previous transactions. Users can create a new account from the web site and these details are stored on an LDAP server on the internal network.

All web server access is via HTTP and all customer transactions are handled via HTTPS. When a customer wants to purchase any fortune cookies they are transferred to a HTTPS session and must first authenticate to the system. All web servers are Sun ONE Web Server 6 running on a Solaris platform. The web servers communicate to the internal LDAP server via an LDAP proxy located on the application server screened subnet. Once authenticated a customer can browse the online catalogue for fortune cookie sayings and store these in a shopping cart session. This user session is managed by the IBM Websphere J2EE application servers, which provide all of the backend application processing, user session tracking and database interaction. The use of Websphere enables each customer to have their own personalised settings retrieved once they have authenticated. The user can then select to purchase the fortune cookie sayings and is required to enter their credit card details, which are validated before the transaction is completed. All user transactions and fortune cookie sayings are stored in an Oracle database, enabling each customer to review their transaction history.

An IBM Websphere J2EE platform was chosen as it offers a flexible and rapid development environment and JAVA is an inherently more secure platform than traditional C/C++ type environments. Websphere application server 5 has been used for the GIAC system.

One feature of the GIAC site is the ability to identify the language setting of the user browser and then direct them to web content in that language. Currently supported languages are:

- English
- French
- German
- Spanish
- Chinese

GIAC additionally provides content transformation for the client device type, whether it be a pc browser (HTML), PDA (cHTML), XML device or WAP (wXML) device. This ensures that GIAC provides content formatted specifically for the end device type.

GIAC is monitoring the success of this approach and is actively working with selected partners to increase the scope of language translations. This has produced increased revenue streams for GIAC through banner advertising in different languages and enabled GIAC to target a larger customer base.

2.2 Suppliers and Partners

Suppliers and partners have access to a supplier and partner portion of the web site which requires a user account name and password for access, which is then authenticated against the LDAP database. This enables suppliers and partners to update their company and contact information, review contracts, track financial details and review their transaction history which is stored on an Oracle database.

GIAC have decided that all suppliers and partners are to utilise a separate IPSEC based VPN connection for any fortune cookie processing. GIAC did not want to use the web site for this purpose as the potential exists for new cookie sayings to be compromised before they are reviewed and saved to the Oracle database.

2.2.1 Suppliers

Suppliers are contracted to provide a specified number of new fortune cookies per week, but are permitted to upload new fortune cookies only. They are not permitted to retrieve, delete or modify any previously uploaded cookies. The uploading of new fortune cookies is achieved via an IPSEC VPN connection. Suppliers upload new fortune cookie sayings to the FTP server. The GIAC FTP server is located within a separate secure subnet and is configured with a user account for each supplier, which corresponds to a separate directory for each supplier. Suppliers are only permitted to upload files and create new directories; they cannot delete, move, copy or rename files. These files are retrieved and reviewed by GIAC personnel and if approved they are uploaded to the production Oracle database and to a replica Oracle database that is used by GIAC partners. Suppliers provide all fortune cookie sayings in English only.

2.2.2 Partners

GIAC partners provide translations for fortune cookies, but do not re-sell them directly. They are resold via the GIAC website with a percentage of the profits going to the partner. Partners can upload and download fortune cookie sayings. The partners use Oracle SQL*Net client to either download fortunes to be translated or upload translated fortunes to a replica Oracle database. Previously supplied and approved fortune cookie sayings from suppliers are loaded to the partner database by GIAC staff for retrieval by GIAC partners. GIAC has supplied some custom SQL queries to each partner to assist in database retrieval and updates.

GIAC fortune cookie processing employees are responsible for retrieving the translated fortune cookie sayings and loading them onto the production database.

Partners utilise the same site to site VPN service as suppliers for access to the GIAC network, but are assigned to separate IP address pools.

2.3 Internal Staff

Internal staff members have unrestricted internet access for web browsing and FTP. All outbound Internet connections are sent via a Netscape proxy server located in a secure subnet. The proxy server handles all HTTP, HTTPS and FTP connections.

2.3.1 EMAIL

Internal email access is via a Microsoft Exchange 2000 server located on the internal network. All external email is sent and received via a Sendmail email server located in a separate screened subnet with other publicly available servers. This is a Linux Red Hat 7.3 server running the latest version of Sendmail. This server performs virus scanning on all emails. All internal to external email traffic is passed between the Exchange server and Sendmail server. Internal users cannot access the Sendmail server and external mail cannot pass directly to the Exchange server.

2.3.2 DNS

A split DNS system is deployed – outbound DNS requests for staff are not sent directly to the Internet, but to an internal DNS server that then perform recursive lookups via the external DNS server.

The external server will be non-recursive, except for all publicly addressable servers. The external DNS is the primary DNS server for the GIAC zone. Two secondary servers are located at different ISPs and zone transfers (TCP 53) are restricted to these secondary DNS servers only. This prevents attackers using zone transfers to learn information about the GIAC network.

Zone transfers are prohibited between the internal and external DNS servers and the external DNS server does not hold any information of the internal systems. The GIAC DNS information in the APNIC database has been kept to a minimum to restrict the useful information that hackers may attempt to gather from APNIC.[1]

The external and internal DNS servers are running BIND version 9.2.1

2.3.3 Internal SOE

All internal employees run Windows 2000 Professional that is loaded with a GIAC standard Operating Environment (SOE). This SOE includes windows hardening, the latest version of Norton anti-virus software and Tiny Personal Firewall with a preconfigured firewall rule base. A software version and patch management process is in place to allow dynamic software updates on a regular basis.

2.3.4 Internal IT Admin Staff

System administration employees do not have any additional access rights through the security infrastructure. All security infrastructure systems can only be accessed from the secure network management server. Whilst this is a

[1] As detailed in SANS Track2 – Firewalls, Perimiter Protection and VPNs, section 2.6 Network Design and assessment

logistical inconvenience for administration staff, it ensures that network access to the critical security equipment is restricted.

2.3.5 Fortune Cookie Processing Staff

One group of internal users is responsible for retrieving the supplier and partner fortune cookies from the secure FTP server and partner Oracle database server. These staff use FTP and Oracle SQL*Net to connect to both the supplier FTP server and partner Oracle database server and the GIAC production Oracle database.

2.4 Mobile Staff and Teleworkers

Mobile staff and home based teleworkers access the GIAC network via a VPN connection utilising a PKI two factor authentication mechanism.

This utilises a Cisco VPN concentrator and Cisco VPN client software, which also provides a personal firewall on each client machine. Aladdin USB tokens are used to store the user's digital certificates and Verisign is used as the trusted third party to host the GIAC certificate authority (CA).

2.5 Defence in Depth Approach

GIAC advocate a 'defence in depth' approach as security is not just a single solution, but many layers, including operational procedures, staff training and staff awareness.

The proposed design solution for GIAC is based on this 'Defence in Depth' approach and will provide GIAC with a highly available network and layered security infrastructure.

Each layer of the infrastructure provides a layer of defence and whilst an intruder may be able to circumvent one layer, the combination of multiple layers adds significant cost to the intruder in terms of effort required, making it very difficult for the security infrastructure to be penetrated.

2.5.1 External Filtering Routers

The external routers will be hardened and a layer of packet filtering applied. This approach will eliminate much of the 'garbage' and more common attack traffic from entering the GIAC network. The routers and ISP links will be configured to provide a high availability and load balanced Internet connection. Two Cisco 3725 routers running Cisco IOS version 12.2.15T will be used as the external ISP facing routers; with each router connecting to a separate ISP. These routers will be hardened as per the NSA 'Router Security

Configuration Guide'[2] and will have some additional filters to prevent RFC1918 and RFC 1466 (IANA IPV4 unassigned addresses[3]) source IP addresses. The Exterior Border Gateway Protocol (eBGP) will be used between the external routers and the ISP routers and Interior Border Gateway Protocol (iBGP) will be used between the two ISP facing external routers. This enables GIAC to advertise BGP reachability information and provides some flexibility in how we want to route our inbound and outbound traffic. Whilst not an immediate requirement, we have decided to configure BGP from day one so that these options are readily available should they be required in the future. BGP filters have been configured to restrict advertised and received routes and to prevent GIAC from being used as transit network by the ISPs. This is covered in detail in the security policy section. In order for the two routers to establish iBGP connections OSPF will be used between these routers as our Interior Gateway Protocol (IGP). They routers will be configured to use OSPF MD5 router authentication.

2.5.2 Switch Hardening

All layer 2 switches will be hardened and the use of secure private VLANs will be employed, providing another layer of defence. Cisco 2950-12 Ethernet switches will be used. All Ethernet switches will have a hardening template applied and telnet access will be restricted to the network management station only with the use of Access control lists.

2.5.3 Server Hardening

All servers will be hardened as much as possible, with all non essential services on each server disabled. All application packages will be installed as non-root users where possible. One of the best tools for securing Solaris is YASSP[4] (http://www.yassp.org) and to harden Linux configurations there are scripts available from http://www.bastille-linux.org.

2.5.4 TripWire[5]

Tripwire will be used to take a snapshot of all binary and configuration files on each server. Periodic checks will be run on each server, with the result compared to previous run. If any files have been altered Tripwire will identify these. This will notify us of any unauthorised files on the system, which may indicate that the server has been compromised and a root kit installed. This is a very valuable defence mechanism.

[2] The National Security Agency (NSA) Router Security Configuration Guide, version 1.0j November 21, 2001

[3] http://www.iana.org/assignments/ipv4-address-space

[5] http://www.tripwire.com

2.5.5 Patch Management Programme

A patch management programme will be put in place to aid the rapid testing and deployment of any OS and application patches relevant to GIAC. If a new vulnerability is announced, GIAC want to have a system in place that allows for rapid deployment of the update.

2.5.6 General Security Topology

The general structure of the network topology ensures that each resource is isolated as much as possible and that only the required traffic is permitted through the firewalls to that host. This approach restricts as much as possible the chance of one compromised host being used to launch an attack against another host. For example if the web server was compromised the firewall rules would not permit a connection from the web server to the VPN concentrator.

2.5.7 VPN Termination

IPSEC based VPN connections will be terminated on the external side of the PIX firewalls, ensuring that all IPSEC authenticated sessions then have to pass through both layers of firewalls. Site to Site VPN using pre shared keys will be used for partner and supplier vpn's. Two factor authentication using digital certificates will be used for remote user vpn connections. A Cisco 3015 VPN concentrator will be deployed.

2.5.8 External PIX Firewalls

The PIX firewalls will operate in failover mode, ensuring high availability, whilst employing the Cisco industry leading ASA technology to deliver stateful packet inspection. This is the first firewall layer in the defence model.

GIAC already has a pair of PIX 525 Firewalls licensed as PIX525-UR and PIX525-FO bundle. These will be re-used in the new design. Six interfaces will be utilised on the PIX 525s for the DMZ, transit and multiple screened subnets that are required. The firewalls will be configured to operate in stateful failover mode. The PIX firewalls will run OS version 6.2(2) which has the Common Criteria Evaluation Assurance Level 4 (EAL4) certification.

2.5.9 Content Switches

Cisco Content Switches have a primary purpose of load balancing traffic between servers, but have a range of useful security features[6]. These include using NAT to hide the real server IP addresses, url filtering; for example

[6] http://www.cisco.com/en/US/customer/products/hw/contnetw/ps789/products_white_paper09186a00800921a6.shtml

filtering nimda and code red, denial of service prevention features and flash crowd protection. Cisco 11503 content switches with integrated SSL offload modules will be deployed.

2.5.10 SSL Offload

SSL offload, in addition to improving performance and reducing server load, adds the ability to inspect otherwise encrypted traffic before it gets to the servers. Once an SSL stream has been decrypted by the offload appliance, the traffic is passed to the servers in a separate connection as HTTP in clear text. This enables IDS to inspect the traffic for any malicious content.

2.5.11 Checkpoint / Nokia Firewalls

The second layer of firewalls – an alternate firewall vendor provides extra protection should a vulnerability be discovered in one of the vendors products. Checkpoint is an industry leading firewall solution providing stateful inspection. Deploying Checkpoint on Nokia appliances adds additional security through the Nokia IPSO platform and high availability.

A pair of Nokia 380s running Checkpoint secure platform will be used for the internal firewalls. These will have a transit link to the PIX firewalls, a screened subnet for the Websphere application servers, a screened subnet for the Oracle database, a screened subnet for the management network and a connection to the internal network.

2.5.12 Intrusion Detection

Intrusion detection systems can recognise various attack signatures embedded within the packet payload and connection streams, which a firewall may not be able to detect. IDS provide another layer in the defence model. GIAC have decided that they will deploy some SNORT sensors at locations in the network. They realise that to deploy and mange IDS effectively is a resource intensive task and are looking at several managed service solutions. In the interim they will re-deploy some surplus servers as IDS sensors and run the latest version of SNORT. They have a good skill level internally to accomplish this.

2.5.13 Virus Scanning

Another layer in the defence, virus scanning ensures that any malicious content, programs, email attachments are identified and isolated before they reach the end hosts.

2.5.14 Secure Network Management

Firewall management servers, Intrusion Detection management servers and logging servers are vital to the operation of the company's security infrastructure. For this reason they should be placed in a highly secure portion of the network. All router, firewall and VPN concentrator login authentication will be managed by a local TACACS database on each device. GIAC plan to implement a RADIUS server at a later stage to manage all device authentications.

2.5.15 NTP

GIAC want to ensure that all system logs have a common timestamp as this will make it easier to correlate events across different systems. To achieve this and provide a level of redundancy, GIAC have deployed two dedicated NTP servers located on their internal network. These are also used by all internal systems. These are Solaris platforms running the latest version of XNTPD.

2.5.16 Syslog

GIAC see the logging of syslog messages as a critical component and have decided to deploy two Linux servers running the latest version of Syslog. These will be attached to a pair of redundant Cisco 2950 Ethernet switches hanging off of the Checkpoint firewalls to ensure that there is no single point of failure. These will be located in a dedicated management screened subnet.

2.5.17 Sink Hole Router

Some self replicating worms generate random IP addresses as the next target systems and these addresses often fall into the RFC1918 and RFC1466 range. A sink-hole router is configured to advertise these address ranges as reachable to internal hosts. The router is configured to send syslog notifications for any connection attempts to these illegal address ranges. This will raise an alarm that an internal host has been compromised and will aid the system administrators in identifying compromised hosts and taking additional preventive or remediation work. Whilst a sink-hole router offers no direct protection, it is a cheap and valuable tool to identify against these kinds of attacks; which may highlight a zero day exploit. This is yet another layer of our 'Defence in Depth' approach. GIAC will deploy an old Cisco 2503 router as a sink-hole router. The role of this router is simply to advertise RFC1918 and RFC 1466 address internally and generate syslog messages against an access list matching these.

All of the above combine to form a defence in depth approach to deploying a security infrastructure.

2.5.18 General DMZ Topology

The network topology is shown in the following diagram.

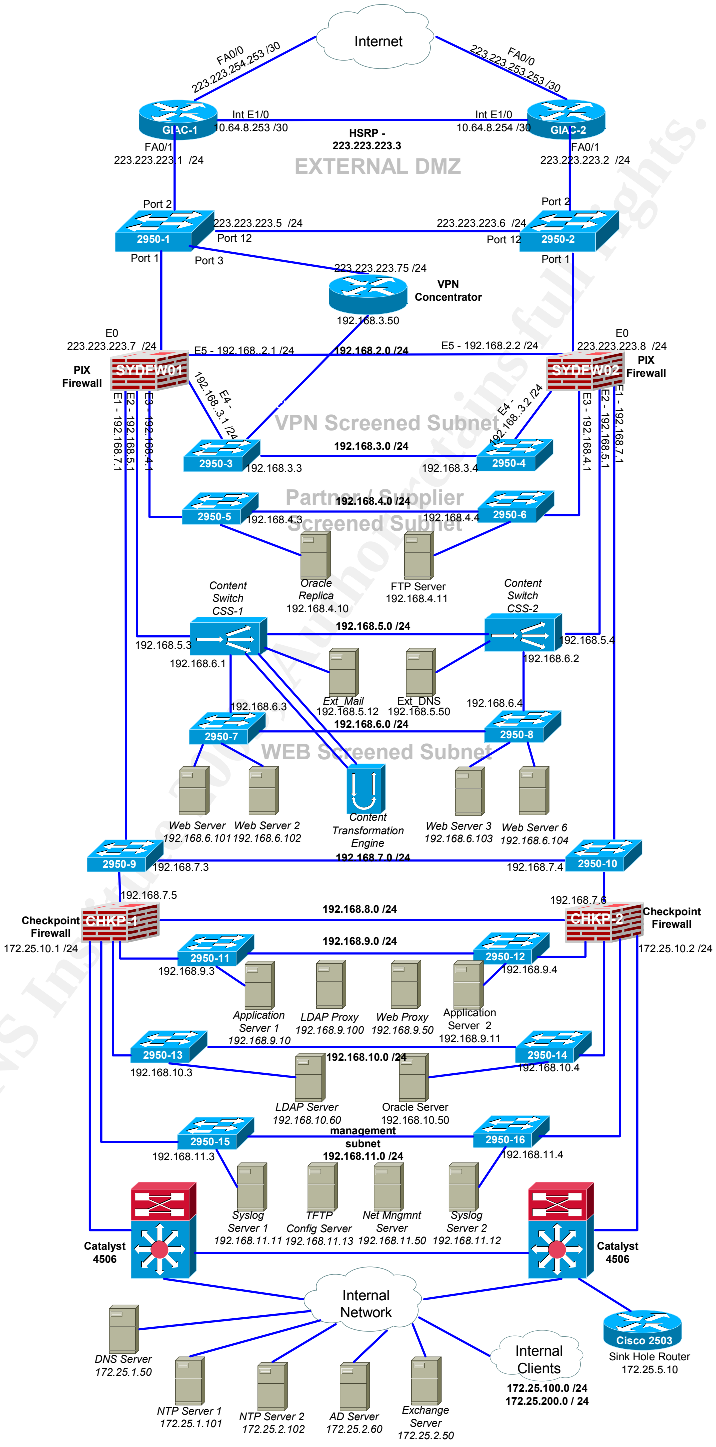


Diagram 1 – Overview of Security Topology

2.6 VPN

A Cisco 3015 VPN concentrator running software version 3.6.x will be used to provide the remote access VPN for remote staff, partners and suppliers. This was chosen because it is a well known platform backed up by excellent support from Cisco. It can be scaled up as demand grows and GIAC have employees that are very familiar with the product.

There will not be any hosts located on the VPN subnet and all access from VPN clients to any hosts must pass through the PIX firewall.

Private VLANs will be configured on the Cisco 2950 switches which will only permit the concentrator to communicate with the PIX Firewall at layer 2.

2.6.1 VPN Concentrator Security

The VPN concentrator will be hardened as much as possible, with all unnecessary features disabled. The concentrator has the capability to define access lists, which will be used to restrict traffic as much as possible.

2.6.2 Supplier / Partner VPN

All supplier and partner VPN connections are site to site Vpn's utilising Cisco routers on the supplier and partner sites and a Cisco VPN 3015 concentrator at GIAC.

The PIX is not used as GIAC want to keep the firewall and VPN functionality separate. In addition to the performance hit that the PIX would suffer performing VPN termination, GIAC also feel that there is more risk of a compromise, and have decided to keep the firewall and vpn functionality in physically separate devices.

2.7 Content Switching

One of GIAC's business goals is to provide a highly available service. To enable this GIAC have decided to deploy Cisco 11503 content switches. This enables traffic to be intelligently load balanced between multiple web servers and provides a highly available environment for GIAC customers, suppliers and partners.

Additional benefits include the ability to take servers down for maintenance and patch upgrades and to seamlessly scale the site by adding more servers as demand grows.
Another feature of the content switches is the ability to perform HTTP header inspection. This is used to identify the language setting of the client browser and direct the user to a web page to one of several languages supported.

Additionally HTTP header inspection is used to identify the end device type. If it is NOT from a PC, the HTTP flow is passed to a content transformation engine. This retrieves the original web content and then dynamically transforms it to a format for the end device (XML, cHTML, WML etc.).

2.8 SSL Offload

GIAC will use SSL offload appliances as this will remove the burden of SSL on the servers and will allow the IDS sensors to inspect the decrypted traffic for malicious content. The SSL offload appliances will be integrated modules in the content switches, which will have rules defined to pass any SSL traffic the SSL offload appliances.

The SSL offload appliance will decrypt the traffic and the initiate a new connection to another VIP address on the content switch. The content switch will then load balance this connection to a real web server.

The ability to inspect SSL traffic in this manner adds another layer of defence.

2.9 IP Addressing Scheme

GIAC have been allocated the public range IP 223.223.223.0[7] /24 and have been allocated their own Autonomous system number (65500)[8] which is required to when multi-homing to different ISPs.

GIAC will be performing NAT on the PIX firewalls to translate their public IP addresses into RFC1918 addressing that is used within their screened subnets and internal network. The IP addressing scheme is shown in the following table.

IP Address Block	Description
223.223.223.0 /24	External public IP range
192.168.2.0 /24	PIX Stateful Failover LAN
192.168.3.0 /24	VPN Screened Subnet
192.168.4.0 /24	Partner / Supplier Server screened subnet
192.168.5.0 /24	Mail, DNS and Web VIPs Screened Subnet
192.168.6.0 /24	Web Server 'Real Addresses' subnet
192.168.7.0 /24	PIX firewall to Checkpoint firewall Transit

[7] This is actually an IANA reserved address range, but is a valid routeable address and is used within this document as an example only.

[8] This is actually a private AS number and is not valid on the Internet. It is used in this document as an example only. Private AS numbers are in the range of 64512 to 65535.

IP Address Block	Description
	subnet
192.168.8.0 /24	Checkpoint Stateful Failover LAN
192.168.9.0 /24	Application server screened subnet
192.168.10.0 /24	Database server screened subnet
192.168.11.0 /24	Management / Logging server subnet
172.25.1.0 /24	GIAC internal servers
172.25.2.0 /24	GIAC internal servers
172.25.5.0 /24	GIAC internal range – used for sink hole router
172.25.100.0 /24	Internal GIAC users
172.25.200.0 /24	Internal GIAC users
10.10.1.0 /24	VPN Pool 1 – GIAC General Users
10.10.2.0 /24	VPN Pool 2 – GIAC IT Admin Staff
10.10.3.0 /24	VPN Pool 3 – GIAC Fortune Cookie Processing
10.10.10.0 /24	Suppliers VPN Pool
10.10.20.0 /24	Partners VPN Pool
10.64.8.252 /30	OSPF Crossover Link between external routers
10.64.8.1 /32	External Router 1 Loopback address
10.64.8.2 /32	External Router 2 Loopback address
223.223.253.252 /30	Link to ISP1
223.223.254.252 /30	Link to ISP2

Table 1 – IP Addressing Scheme

3. Security Policy and Tutorial

This section details the configuration and security policy of the Border routers, the VPN concentrator and a tutorial of the primary PIX firewalls configuration.

3.1 External Border Routers

The following details the external router configuration.
The routers will be hardened as per the NSA guidelines.[9] Below is a brief description of the protocols and services that will be disabled. Depending on the IOS version is use, some of these services are already disabled by default, but it is good practice to explicitly turn these off to be sure. Please refer to the NSA guide or the numerous sources available on the Internet and at http://www.cisco.com for more details.

3.2 General Hardening

All unused services will be explicitly disabled. The hardening commands listed below are to be performed from global configuration mode, unless explicitly stated as an interface configuration command.

3.2.1 Disable Cisco Discovery Protocol (CDP)

Cisco Discovery Protocol (CDP) is a media independent protocol, which is enabled by default on all Cisco devices. It is used for some network management functions, but is dangerous in that it allows any system on a directly connected segment to learn that the router is a Cisco device, and to determine the model number and the Cisco IOS software version being run. This information may be used to design attacks against the router. CDP will be disabled.
To disable CDP, the command required is

```
no cdp run
```

3.2.2 Disable TCP and UDP Small Servers

The TCP and UDP small servers are legacy services running in IOS. These servers run Echo, Chargen, Discard and Daytime services.
These services are infrequently used for legitimate purposes. Since these services are legacy and not required, they will be disabled.

```
no service tcp-small-servers
no service udp-small-servers
```

[9] NSA-Router Security Configuration Guide - http://nsa1.www.conxion.com/cisco/

3.2.3 Disable IP Finger

Finger can be used to find out which users are logged into a network device and this information can sometimes be useful to an attacker.
The finger service will be explicitly disabled.

no ip finger

3.2.4 Disable HTTP Server

The HTTP server enables web based administration of the router. The HTTP server will be explicitly disabled to ensure that any potential web based exploits cannot be run on the router.

no ip http server

3.2.5 Disable BOOTP Server

BOOTP (Bootstrap Protocol) is a protocol that lets a network user/device be automatically configured (receive IP addressing information) and have an operating system boot without user involvement. The BOOTP server service is not required and will be disabled.

no ip bootp server

3.2.6 Disable Configuration Auto loading

The router can load a configuration file from a remote server. This is not secure, so it is disabled.

No boot network
no service config

3.2.7 Disable IP Source Routing

The IP protocol supports source routing options that allow the sender of an IP datagram to control the route that datagram will take. These options are rarely used for legitimate purposes in real networks. Source routing will be disabled.

no ip source-route

3.2.8 Disable Proxy ARP

It is recommended that Proxy ARP be disabled on all interfaces. This will ensure that the router will not answer ARP's for any addresses other than the addresses bound to each interface. This is an interface configurations command

Interface Ethernet x/x

```
no ip proxy-arp
```

3.2.9 Disable Directed Broadcasts

Directed broadcasts enable the router to forward packets destined for a broadcast address to all devices within that broadcast range. Smurf attacks use directed broadcasts by sending an ICMP echo request to a broadcast address, and a router configured with IP directed broadcasts will forward these packets.
IP directed broadcasts will be disabled. This is an interface configuration command.

```
Interface Ethernet x/x
no ip directed-broadcast
```

3.2.10 Disable ICMP Unreachables, Redirects and mask Replies

Attackers often use ICMP 'host unreachable', 'redirect' and 'mask reply' messages for mapping a network. To disable these, enter the following configuration command on each interface of the router.

```
Interface Ethernet x/x
no ip unreachables
no ip mask-reply
no ip redirects
```

3.2.11 Disable SNMP-Server

SNMP is used for monitoring and administration purposes, but GIAC do not require any SNMP management of the external routers, so the SNMP server will be disabled.

```
no snmp-server
```

3.2.12 Disable IP domain-lookup

IP domain lookup is used for name address translation. This service is not required for the external routers, so it will be disabled.

```
no ip domain-lookup
```

3.2.13 Disable service pad

Service pad is a Packet assembly and disassembly service, which allows connection between PAD devices and access servers. This service is not required, so it will be disabled.

```
no service pad
```

3.2.14 Disable DHCP Service

The DHCP service assigns IP addresses to hosts on a specified interface. This service is not required and will be disabled.

```
no service dhcp
```

3.2.15 Enable Password

The Cisco enable password type 7 uses a trivial Cisco encoding mechanism for the password. Only enable secret passwords should be used which use MD5, a one-way cryptographic hash. Cisco Type 7 passwords can be taken from a router configuration and easily decoded using numerous utilities found on the Internet.
It is recommended that only the enable secret password be used on the router. The existing enable password should be removed. The use of enable secret passwords uses MD5 hashing automatically.

```
enable secret <PASSWORD>
no enable password
```

3.2.16 Enable Service password-encryption

The service password-encryption command directs the router to encrypt the router passwords. Password-encryption is a low level encryption method which is used on all passwords except the ones used with enable secret. It isn't designed to be a high level secure password, but it is designed to prevent observers looking over the administrators shoulder at a console.
It is recommended that you use service password-encryption to secure passwords.

```
service password-encryption
```

3.2.17 Login Banner

An exec and message of the day login banners will be set on each router to warn against any unauthorised access.

```
banner exec ^CCCC
****************************************************************
****************************************************************
                      WARNING
Access to this system is for authorised users and for authorised
purposes only.
Unauthorised access or use is a serious breach of security policies.
For staff this may involve disciplinary action up to and including
dismissal, it may also be a criminal or civil offence.
If you or your intended use are not authorised do not proceed to log on
```

```
to this system.
************************************************************
*******************************************************************^C
banner motd ^CCCC
************************************************************
************************************************************
This network, information about its components and information systems
within it are CONFIDENTIAL. Access to, or use of, this network by
unauthorised people (including subsidiary companies and their
personnel)
or for any other unauthorised purpose is STRICTLY PROHIBITED.
This Router records and logs user IP addresses.
*******************************************************************
******************************************************************* ^C
```

3.2.18 NTP

The Network Time Protocol (NTP) provides synchronized time for devices. When enabled, NTP enables you to standardise and synchronise time for all devices to a centralised time source. It is important to explicitly configure a trusted time source, and to use proper authentication, since corrupting the time base is a good way to subvert certain security protocols. The Cisco implementation of NTP supports cryptographic authentication using MD5.
NTP does not pose a high security risk as a service, however, when you are logging from a number of routers, it is recommended that you use a common secure time source for all routers.

```
ntp authentication-key 40 MD5 <secretkey>
ntp authenticate
ntp soure loopback0
ntp server 172.25.1.101
ntp server 172.25.2.102
```

3.2.19 SYSLOG

Syslog can provide information on the state of the router, from informational messages right through to debugging messages. Syslog can also be referenced when troubleshooting an OS or Hardware related problems.
Syslog will be enabled to send messages to the internal syslog servers on the management screened subnet. Syslog will be configured to use an internal loopback interface as the source address. This ensures that should the internal facing Ethernet interface or Ethernet switch fail, the router can still send syslog messages to the syslog server via an alternate path through the second router. Syslog will be set to log at local5 level during normal operation, as other higher levels of logging and may cause unnecessary loads on the router will create unnecessary syslog messages. Higher levels of logging can be enabled manually as required.

```
logging buffered 16384 debugging
logging history notifications
Logging on
No logging console
logging trap errors
logging facility local5
logging source-interface loopback0
logging 192.168.11.11
logging 192.168.11.12
```

3.2.20 Timestamps

Timestamps can be added to provide extra information to logs and debugging events. Enabling a more granular timestamp will ensure that more detail will be logged for any of the events which have occurred.

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

3.2.21 Configure AAA Access Control and Accounting with TACACS+

AAA Security Services provides a modular way of performing authentication, authorization and accounting for users accessing the router. The commands listed below will be used to identify users, authenticate them via the use of a local TACACS+ database and allow administration. In addition to authenticating and assigning access to the users via the 'aaa authentication' command, the AAA accounting command tracks and logs user activity and access to the routers.
The username and password below should be comprised of upper and lowercase characters. This will make it more secure and will increase the cost in terms of effort for brute force attacks against the routers.
As part of best practice implementation AAA will be used. This requires a valid username / password to be entered for access to the router, rather than just a password.
The AAA configuration currently uses a local database on the router, but GIAC plan implement a TACACS+ or Radius server at a later date.

```
aaa new-model
aaa authentication login GIAC local
username user1 password xxxxxxxx
line con 0
 login authentication GIAC
```

3.2.22 Apply Access-Lists to VTY (Telnet) Sessions

It is recommended that VTY access only be permitted from specific hosts. An access-list needs to be created to only allow VTY access from those hosts. Logging should also be enabled on this access list.
The commands to restrict access to VTY must be performed from both global configuration mode and on the VTY 0 4 interfaces. In the commands below, the IP address 192.168.11.50 is the IP address of the network management station and 10.64.8.1 is the loopback address of external router GIAC-1.

```
access-list 100 remark VTY Access List
access-list 100 permit tcp host 192.168.11.50 host 10.64.8.1 eq 23 log-
input
access-list 100 deny ip any any log-input
line vty 0 4
access-class 100 in
```

3.2.23 Console Inactivity Timeout

It is recommended that an inactivity timer be set on the console and vty lines. This prevents a logged in user session from being indefinitely active, which would be a security threat. The console port will have the further restriction of being unable to initiate any additional connections.

```
line con 0
 session-timeout 5
 exec-timeout 5 0
 login authentication GIAC
 transport output none
line vty 0 4
 session-timeout 5
 exec-timeout 5 0
 login
```

3.2.24 Disable AUX port access

The auxiliary port is not required, so all access via this port will be explicitly disabled.

```
Line aux 0
 No exec
```

3.2.25 BGP Configuration

The Border Gateway Protocol (BGP) version 4 will be used between the external routers and the ISP routers. Initially the ISPs have been requested to only advertise a default route via BGP. A multihomed BGP configuration requires that GIAC have a BGP Autonomous system (AS) number. Luckily, someone at GIAC had the foresight to request an AS number several years ago, so this requirement is already in place.[10]

GIAC will be multihomed to ISPX1 and ISPY1, with ISPX1 being the primary for outbound traffic and ISPY1 being the standby for outbound traffic. Inbound traffic will be shared over both links.
BGP is a very flexible IP routing protocol that has extensive capabilities to allow the configuration of various routing policies and traffic control characteristics. Some of these BGP capabilities will be used to achieve the required traffic flow required by GIAC. These are outlined in the following sections.

3.2.26 Outbound Traffic

Both of the external routers will have their BGP weight attribute set so that they always prefer their neighbouring ISP router as the next hop for their Default route. HSRP will be used between the routers, with the HSRP address being used as the next hop for the default route on the PIX firewalls. The external router GIAC-1 which peers to ISPX1 will be configured as the primary HSRP router. This ensures that under normal operation all outbound traffic will be via the link to ISPX1.

3.2.27 Transit Area Prevention

It is important when multihoming to different ISP's, that GIAC does not become a transit area. This would allow traffic from ISPX1 to ISPY1 (for example) and vice versa to traverse GIAC's AS. Although ISPs should not send traffic via their customers, the possibility does exist, which could result in a serious denial of service type scenario.

To prevent this transit routing, a feature called BGP as-path filtering has been configured on the GIAC routers with an outgoing BGP filter list. This will ensure that only the GIAC AS number and routes are advertised to the ISPs.
Inbound filters have been used to ensure that GIAC only receives a default route the ISPs.

Internal BGP (iBGP) will be used between the two GIAC external routers to provide dynamic re-routing should either of the ISP links fail. BGP timers have been reduced from the default values to speed up failover.

The BGP configuration for the external router GIAC-1 is shown below. A similar configuration is applied to the second external router GIAC-2. A full router configuration of GIAC-1 is listed in Appendix-C.

[10] More information can be obtained from the APNIC at http://www.apnic.net/db/AS.html

GIAC-1 BGP Configuration

```
router bgp 65500
 no synchronization
 bgp log-neighbor-changes
 network 223.223.223.0 mask 255.255.255.0
 neighbor 10.64.8.2 remote-as 65500
 neighbor 10.64.8.2 description IBGP to GIAC-2
 neighbor 10.64.8.2 update-source Loopback0
 neighbor 10.64.8.2 timers 5 15
 neighbor 10.64.8.2 next-hop-self
 neighbor 10.64.8.2 default-originate
 neighbor 10.64.8.2 soft-reconfiguration inbound
 neighbor 223.223.254.254 remote-as 64551
 neighbor 223.223.254.254 description EBGP to ISPX1
 neighbor 223.223.254.254 distribute-list 1 out
 neighbor 223.223.254.254  route-map default_route_preference in
 neighbor 223.223.254.254 filter-list 2 out
 neighbor 223.223.254.254 prefix-list ISPX1_Default in
 no auto-summary
!
ip as-path access-list 2 permit ^65500$
ip as-path access-list 2 permit ^$
ip as-path access-list 2 deny .*
!
ip prefix-list ISPX1_Default seq 5 permit 0.0.0.0/0
!
route-map default_route_preference permit 10
 match ip address prefix-list ISPX1_Default
 set weight 100
!
```

3.2.28 HSRP

HSRP is used between the two external routers. The router (GIAC-1) is configured as the primary router for the HSRP address and the PIX firewalls are configured with a default route via the HSRP address. The HSRP priority on this router is set to 120, from the default 100.

HSRP interface tracking has been configured, so that if the GIAC-1 router's ISPX1 facing interface goes down, the HSRP priority will be decremented by a value of 30 to 90. This will enable the secondary router (GIAC-2) to become the primary router for the HSRP address and all traffic will transparently be directed from the Firewall directly to the GIAC-2 router. HSRP timers have been tuned down from the default values to speed up failover.

GIAC-1 HSRP Configuration

```
interface FastEthernet0/1
 description ------ DMZ LAN ------
 bandwidth 100000
 ip address 223.223.223.1 255.255.255.0
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 speed 100
 full-duplex
 no cdp enable
 standby 1 ip 223.223.223.3
 standby 1 timers 2 5
 standby 1 priority 120
 standby 1 preempt
 standby 1 track FastEthernet0/0 30
```

GIAC-2 HSRP Configuration

```
interface FastEthernet0/1
 description ------ DMZ LAN ------
 bandwidth 100000
 ip address 223.223.223.2 255.255.255.0
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 speed 100
 full-duplex
 no cdp enable
 standby 1 ip 223.223.223.3
 standby 1 timers 2 5
 standby 1 preempt
```

3.2.29 OSPF Configuration

To provide connectivity for the Internal BGP (iBGP) speakers, an interior routing protocol is required. The OSPF IP routing protocol has been deployed in this design as the interior routing protocol. OSPF is a link state protocol that allows for fast dynamic convergence to any topology changes.

OSPF has been configured on the loopback interfaces and on the Ethernet crossover link between the two routers. The iBGP peers have been configured to use their loopback address as the iBGP source and the opposing routers loopback address as the iBGP peer destination address.
Using OSPF provides a way to implement redundancy should the internal Ethernet (router interface or Ethernet switch) to the DMZ fail. If this occurs, OSPF will dynamically re-converge and will redirect traffic over the crossover link.

For example, if the Ethernet interface on router GIAC-1 to the PIX Firewall fails, traffic from this router destined to the Firewall will automatically be rerouted over the crossover link to router GIAC-2 and then forwarded to the Firewall. When the interface recovers, OSPF will automatically converge again and traffic will go directly from GIAC-1 to the Firewall.

OSPF and route maps have been configured to redistribute the DMZ network into OSPF under such a failure scenario and the MD5 hashing algorithm has been configured between OSPF neighbours to ensure that all OSPF traffic is encrypted.

GIAC-1 OSPF Configuration

```
router ospf 100
 log-adjacency-changes
 redistribute connected subnets route-map Into_OSPF
 network 10.0.0.0 0.255.255.255 area 0.0.0.0
 default-metric 100
!
interface Loopback0
 description ------ Loopback Interface for BGP and OSPF ------
 ip address 10.64.8.1 255.255.255.255
!
interface Ethernet1/0
 description ------ Crossover to GIAC-2 ------
 bandwidth 10000
 ip address 10.64.8.253 255.255.255.252
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 ip ospf authentication
 ip ospf authentication-key 7 0506071D2D1C40
 ip ospf priority 2
 full-duplex
 no cdp enable
!
access-list 10 remark Distribute FA0/1 LAN into OSPF on Crossover Link
access-list 10 permit 223.223.223.0 0.0.0.255
!
route-map Into_OSPF permit 10
 match ip address 10
!
```

GIAC-2 OSPF Configuration

```
router ospf 100
 log-adjacency-changes
 redistribute connected subnets route-map Into_OSPF
 network 10.0.0.0 0.255.255.255 area 0.0.0.0
 default-metric 100
```

```
!
interface Loopback0
 description ------ Loopback Interface for BGP and OSPF ------
 ip address 10.64.8.1 255.255.255.255
!
interface Ethernet1/0
 description ------ Crossover to GIAC-1 ------
 bandwidth 10000
 ip address 10.64.8.254 255.255.255.252
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 ip ospf authentication
 ip ospf authentication-key 7 060B0E33401E07
 full-duplex
 no cdp enable
!
access-list 10 remark Distribute FA0/1 LAN into OSPF on Crossover Link
access-list 10 permit 223.223.223.0 0.0.0.255
!
route-map Into_OSPF permit 10
 match ip address 10
!
```

3.2.30 Miscellaneous Configuration

All unused interfaces will be explicitly disabled.

```
Interface x/x
Shutdown
```

All interfaces will have their speed and duplex explicitly hard coded to 100Mbps full-duplex. No auto negotiation will be permitted.

```
Interface x/x
Speed 100
Duplex full
```

3.2.31 Internal routing

Additional routing on the external routers is not required. The routers need to communicate to the internal syslog and NTP servers, but will send traffic destined for these servers to IP addresses in the 223.223.223.0 /24 range. The PIX will then perform NAT to the internal addresses. This use of NAT negates the requirement to configure any internal routes on the external routers.

As the source IP address will be from the 10.64.8.0 /24 range the PIX will require a route back for return traffic. The PIX will have a default route via the external routers that will accomplish this.

3.2.32 Ingress and Egress filtering.

The external GIAC routers will be configured to perform packet filtering with the objective of preventing as much 'garbage traffic' as possible from reaching the firewalls. This will be accomplished with the use of extended ACL's which offer more extensive filtering capabilities over standard ACL's.

ACL's will be applied inbound on both the external ISP facing interface and the internal DMZ facing interface. This method of filtering is used to ensure that traffic is blocked before it passes through the router.

Ingress and egress filtering ensures that no RFC1918, RFC1466 (reserved IANA address space) or broadcast and multicast source addresses will be routed inbound or outbound.

Ingress filtering will ensure that any spoofed packets will be dropped by the perimeter router, while egress filtering will ensure that any traffic from misconfigured devices behind the Internet routers (or spoofed by internal devices) will not be sent out to the Internet.

The following define the access lists. Lines beginning with a '!' are comments and the 'log' keyword tells the router to generate a syslog message when an access list match occurs.

Ingress Filter

```
Interface FastEthernet 0/0
  Ip access-group 199 in
!
access-list 199 remark From internet Filter

! Block Loopback, reserved, RFC 1918 and RFC 1466 addresses
access-list 199 deny   ip 127.0.0.0 0.255.255.255 any log
access-list 199 deny   ip 0.0.0.0 0.255.255.255 any log
access-list 199 deny   ip 1.0.0.0 0.255.255.255 any log
access-list 199 deny   ip 2.0.0.0 0.255.255.255 any log
access-list 199 deny   ip 10.0.0.0 0.255.255.255 any log
access-list 199 deny   ip 23.0.0.0 0.255.255.255 any log
access-list 199 deny   ip 31.0.0.0 0.255.255.255 any log
access-list 199 deny   ip 67.0.0.0 0.255.255.255 any log
access-list 199 deny   ip 68.0.0.0 3.255.255.255 any log
access-list 199 deny   ip 72.0.0.0 3.255.255.255 any log
access-list 199 deny   ip 80.0.0.0 15.255.255.255 any log
```

```
access-list 199 deny   ip 96.0.0.0 15.255.255.255 any log
access-list 199 deny   ip 112.0.0.0 3.255.255.255 any log
access-list 199 deny   ip 126.0.0.0 1.255.255.255 any log
access-list 199 deny   ip 169.254.0.0 0.0.255.255 any log
access-list 199 deny   ip 172.16.0.0 0.15.255.255 any log
access-list 199 deny   ip 191.255.0.0 0.0.255.255 any log
access-list 199 deny   ip 192.0.2.0 0.0.0.255 any log
access-list 199 deny   ip 192.168.0.0 0.0.255.255 any log
access-list 199 deny   ip 198.18.0.0 0.0.255.255 any log
access-list 199 deny   ip 201.0.0.0 0.255.255.255 any log
access-list 199 deny   ip 222.255.255.0 0.0.0.255 any log
access-list 199 deny   ip 223.255.255.0 0.0.0.255 any log

! Block our own range as a source (this range is used an an example
only and would normally be blocked as shown as the last line above)
access-list 199 deny   ip 223.223.223.0 0.255.255.255 any log

! Block and log Multicast source addresses
access-list 199 deny   ip 224.0.0.0 31.255.255.255 any log
! Permit specific traffic destined to our available services
access-list 199 permit tcp any host 223.223.223.100 eq 80
access-list 199 permit tcp any host 223.223.223.100 eq 443
access-list 199 permit tcp any host 223.223.223.50 eq 53
access-list 199 permit udp any host 223.223.223.50 eq 53
access-list 199 permit udp any host 223.223.223.12 eq 25


!Permit IKE, ESP for partner / supplier and IPSEC over TCP 10000 for
GIAC VPN clients
access-list 199 permit udp any host 223.223.223.75 eq 500
access-list 199 permit esp any host 223.223.223.75
access-list 199 permit tcp any host 223.223.223.75 eq 10000


!Permit BGP from our upstream ISP router
access-list 199 permit tcp host 223.223.254.254 host 223.223.254.253
eq bgp


! pemit replies to connections initiated from the inside. This allows any
packets through that DO NOT have the SYN flag set. This can be easily
bypassed, but does offer some basic protection.
access-list 199 permit tcp any any established


! We do not want to block all ICMP as some ICMP traffic is required. We
permit required ICMP traffic.
access-list 199 permit icmp any 223.223.223.0 0.0.0.255 echo-reply
```

```
access-list 199 permit icmp any 223.223.223.0 0.0.0.255 host-unreachable
access-list 199 permit icmp any 223.223.223.0 0.0.0.255 net-unreachable
access-list 199 permit icmp any 223.223.223.0 0.0.0.255 packet-too-big


! Deny and log everything else
access-list 199 deny ip any any log
```

Note: We would normally block all 223.255.255.0 0.0.0.255 traffic, but we have used the 223.223.223.0 /24 range as our sample GIAC IP address range.

Egress Filter

```
access-list 198 remark To internet Filter

! Block Loopback, reserved, RFC 1918 and RFC 1466 addresses
access-list 198 deny   ip 127.0.0.0 0.255.255.255 any log
access-list 198 deny   ip 0.0.0.0 0.255.255.255 any log
access-list 198 deny   ip 1.0.0.0 0.255.255.255 any log
access-list 198 deny   ip 2.0.0.0 0.255.255.255 any log
access-list 198 deny   ip 10.0.0.0 0.255.255.255 any log
access-list 198 deny   ip 23.0.0.0 0.255.255.255 any log
access-list 198 deny   ip 31.0.0.0 0.255.255.255 any log
access-list 198 deny   ip 67.0.0.0 0.255.255.255 any log
access-list 198 deny   ip 68.0.0.0 3.255.255.255 any log
access-list 198 deny   ip 72.0.0.0 3.255.255.255 any log
access-list 198 deny   ip 80.0.0.0 15.255.255.255 any log
access-list 198 deny   ip 96.0.0.0 15.255.255.255 any log
access-list 198 deny   ip 112.0.0.0 3.255.255.255 any log
access-list 198 deny   ip 126.0.0.0 1.255.255.255 any log
access-list 198 deny   ip 169.254.0.0 0.0.255.255 any log
access-list 198 deny   ip 172.16.0.0 0.15.255.255 any log
access-list 198 deny   ip 191.255.0.0 0.0.255.255 any log
access-list 198 deny   ip 192.0.2.0 0.0.0.255 any log
access-list 198 deny   ip 192.168.0.0 0.0.255.255 any log
access-list 198 deny   ip 198.18.0.0 0.0.255.255 any log
access-list 198 deny   ip 201.0.0.0 0.255.255.255 any log
access-list 198 deny   ip 223.255.255.0 0.0.0.255 any log

! Block and log Multicast source addresses
access-list 198 deny   ip 224.0.0.0 31.255.255.255 any log

! Block and log any outbound netbios traffic
access-list 198 deny tcp any any range 135 139 log
access-list 198 deny tcp any any eq 445 log
```

```
! Deny certain ICMP traffic as ths could be used to map the GIAC network.
access-list 198 deny icmp any any time-exceeded
access-list 198 deny icmp any any host-unreachable
access-list 198 deny icmp any any echo-reply

!Permit traffic sourced from the PIX outside interface, IPSEC and http from the VPN concentrator only
access-list 198 permit ip host 223.223.223.7 any
access-list 198 permit udp host 223.223.223.75 any eq 500
access-list 198 permit esp host 223.223.223.75 any
access-list 198 permit tcp host 223.223.223.75 eq 10000 any
access-list 198 permit tcp host 223.223.223.75 host <verisign IP Address>  eq 80

! Deny and log everything else
access-list 198 deny ip any any log
```

We have configured a permit statement for our own source range and deny for everything else, but we want to log any events for the specific traffic defined above, such as spoofed source IP addresses and NetBIOS traffic. This provides valuable information about our internal users, any compromised internal hosts and potentially malicious activity from internal users. The above ACL ordering will be reviewed after a period of time and based on the hit rates, will be re-ordered accordingly.

See Appendix C for a full router configuration of GIAC-1.

3.3 PIX Configuration and Tutorial

This is a guide to configuring the PIX with specific details on the GIAC implementation. For a more detailed PIX configuration reference and command guide, refer to the following Cisco document.

http://www.cisco.com/en/US/customer/products/sw/secursw/ps2120/products_command_reference_chapter09186a008010423d.html

3.3.1 Firewall Overview

The external firewalls will be Cisco PIX525s, running as a failover pair. The firewall configuration is explained below.

The firewall topology will result in all traffic flowing through a single firewall, but all session state information will be maintained in the second firewall, through the PIX high-availability feature. Should the primary firewall fail, the

second firewall will assume operation and control of all active flows, maintaining session state.
The PIX firewalls have been configured with six interfaces as follows:

- outside
- inside
- vpn
- part_sup
- web
- sync (Stateful Failover connection)

3.3.2 General Configuration

First we need to set the hostname, system prompt enable password, telnet password and interface speeds.

```
Hostname PIXFW01
Enable password ‘somethinggood’
Passwd ‘somethingelsegood’
interface ethernet0 100full
interface ethernet1 100full
interface ethernet2 100full
interface ethernet3 100full
interface ethernet4 100full
interface ethernet5 100full
```

Note: We want to make sure we hard code both the firewall interfaces and the switch ports that the firewalls will connect to. This will prevent any auto-negotiation problems.
We will use the local TACACS database for user authentication.

```
username admin password ‘somethinggood’ encrypted privilege 15
aaa authentication enable console LOCAL
aaa authentication http console LOCAL
aaa authentication serial console LOCAL
aaa authentication ssh console LOCAL
```

We will add another user and assign monitor privileges only. Then we need to enable command authorisation to enable the different command privilege levels

```
username monitor password 28OsuAeHLWtpOHWa encrypted privilege 3
aaa authorization command LOCAL
```

Next we need to assign meaningful DMZ names and security levels to the physical interfaces of the firewall.

```
nameif ethernet0 outside security0
nameif ethernet1 inside security100
nameif ethernet2 partsub security50
nameif ethernet3 web security50
nameif ethernet4 vpn security20
nameif ethernet5 sync security90
```

The PIX uses security levels to assign a 'trust' level to the interface, where 0 is the least trusted and 100 is the most trusted. As long as a translation rule exists the PIX will allow traffic to pass from a more secure interface to a lower security level interface, but not the reverse. We have set the 'web' and 'partsup' interfaces to the same security level. The PIX will not permit ANY traffic between two interfaces with the same security level.

We need to assign IP addresses to the PIX interfaces as follows.

```
ip address outside 223.223.223.7 255.255.255.0
ip address inside 192.168.7.1 255.255.255.0
ip address partsub 192.168.4.1 255.255.255.0
ip address web 192.168.5.1 255.255.255.0
ip address vpn 192.168.3.1 255.255.255.0
ip address sync 192.168.2.1 255.255.255.0
```

3.3.3 Routing

The PIX knows which local subnets are attached by the IP address assigned to each interface, but for non-local subnets the PIX must have a valid route in its routing table. Static routes will be used for all of the internal networks and a default route for all Internet bound traffic. All static routes will be assigned an administrative distance of 1.

The default route is the 'catch all' route that is matched if there are no specific route table matches in the PIX. The default route for the PIX will be via the outside interface to the HSRP address of the external routers.

```
Route outside 0 0 223.223.223.3 1
```

Set routes for the VPN pools. These are via the VPN concentrator.

```
Route vpn 10.10.1.0 255.255.255.0 192.168.3.50 1
Route vpn 10.10.2.0 255.255.255.0 192.168.3.50 1
Route vpn 10.10.3.0 255.255.255.0 192.168.3.50 1
Route vpn 10.10.10.0 255.255.255.0 192.168.3.50 1
Route vpn 10.10.20.0 255.255.255.0 192.168.3.50 1
```

Define the routes for all other non-local screened subnets and the internal ip address ranges that need to be reachable by the PIX.

```
Route inside 192.168.9.0 255.255.255.0 192.168.7.5 1
Route inside 192.168.11.0 255.255.255.0 192.168.7.5 1
Route inside 172.25.0.0 255.255.0.0 192.168.7.5 1
```

Note that we do not include a route to the database subnet (192.168.10.0 /24). All traffic to the Oracle and LDAP servers must go via the Proxies. For this reason and to maintain our security model, we do not include a direct route from the PIX to the database segment.

For similar reasons we do not include a route to the web server subnet (192.168.6.0 /24). This subnet is behind the content switches, which perform the NAT function from the virtual IP addresses to the real web server addresses. This adds another layer to our security model by 'hiding' the real addresses of the web servers.

We are screening for spoofed source addresses on the external routers, but there is a possibility that this layer can be breached, so we configure some anti-spoofing features on the PIX also.

```
ip verify reverse-path interface outside
ip verify reverse-path interface inside
ip verify reverse-path interface partsup
ip verify reverse-path interface web
ip verify reverse-path interface vpn
ip verify reverse-path interface sync
```

The reverse path features verifies that all packets received on that interface have a source address which is reachable via that same interface – i.e. there is a valid route (or a default route) to that network via that interface. If not the packet is dropped[11]

3.3.4 Failover Configuration

For our design, the PIX will operate as a pair of firewalls in high availability mode. This means we have an active firewall and a failover firewall. The 'sync' interface will be configured as the failover link. This will replicate all 'state' information across this link to the standby firewall.

```
failover
failover timeout 0:00:00
failover poll 3
failover ip address outside 223.223.223.8
failover ip address inside 192.168.7.2
```

[11] A similar feature (unicast reverse path forwarding) is available in BGP, the primary routing protocol of the Internet. If more ISPs applied this feature it would prevent a lot of spoofed traffic from reaching customer networks in the first place! Ask your ISP if they support this feature and try and encourage them to enable it ☺

```
failover ip address partsup 192.168.4.2
failover ip address web 192.168.5.2
failover ip address vpn 192.168.3.2
failover ip address sync 192.168.2.2
failover link sync
```

The failover timeout parameter sets the amount of time, that should a failover occur, the standby unit should pass traffic without requiring a prior xlate to exist.

3.3.5 Admin Access

The PIX will be configured to only accept SSL and HTTPS connections for administration. The PIX has a GUI management application called PIX Device Manager (PDM)[12]. This enables configuration and monitoring from a web browser. Access will be restricted to the internal network management workstation on the management subnet only (192.168.11.50).

```
http server enable
http 192.168.11.50 255.255.255.255 inside
ssh 192.168.11.50  255.255.255.255 inside
ssh timeout 5
```

Information on PDM can be found at:

http://www.cisco.com/en/US/customer/products/sw/netmgtsw/ps2032/products_installation_guide_chapter09186a00800e3826.html

3.3.6 Logging

Logging is a critical component of any security system and is our primary method of alert notification. As with all of our network and security devices we will send all logging information to two syslog servers.

```
logging on
no logging console
logging timestamp
logging standby
logging monitor informational
logging buffered informational
logging trap informational
logging facility 22
logging host 192.168.11.11
logging host 192.168.11.12
```

[12] PDM requires some additional code to be installed on the PIX which is installed as a separate image file. When you connect via HTTPS, a JAVA applet is downloaded to your browser. Pls refer to Appendix E for samples of PDM screenshots.

Logging level 22 will issue informational log messages. These are a normal, but significant condition. Debug level logging (facility 23) will be enabled as required.

3.3.7 NTP and SMNP

NTP will be configured to ensure a consistent time stamp for the PIX and any syslog messages. We do not need SNMP so it will be explicitly disabled.

```
clock timezone AST 11
ntp server 172.25.1.101 source inside
ntp server 172.25.2.102 source inside
no snmp-server
```

3.3.8 Fixup Protocol

The 'fixup protocol' command enables Cisco Adaptive Security Algorithm (ASA) for certain protocols. ASA enables the stateful inspection of certain protocols that are specified in the PIX configuration. By default the PIX enables fixup for FTP 21, HTTP 80, H323 H225 1720, H323 RAS 1718-1719, ILS 389, RSH 514, RTSP 554, SMTP 25, SQL*NET 1521, SIP 5060, SKINNY 2000. An extract from www.cisco.com regarding the SMTP fixup reads:

> *"The **fixup protocol smtp** command enables the Mail Guard feature, which only lets mail servers receive the RFC 821, section 4.5.1, commands of HELO, MAIL, RCPT, DATA, RSET, NOOP, and QUIT. All other commands are translated into X's which are rejected by the internal server. This results in a message such as "500 Command unknown: 'XXX'." Incomplete commands are discarded."*
>
> Source: http://www.cisco.com/en/US/customer/products/sw/secursw/ps2120/products_command_reference_chapter09186a00801049b7.html#42126

The default ports can be changed or additional services can be added. For example another HTTP service listening on port 81, can be added if required. For the GIAC implementation we need the following fixup protocols enabled and will keep the default ports.

```
fixup protocol ftp 21
fixup protocol http 80
fixup protocol ils 389
fixup protocol smtp 25
fixup protocol sqlnet 1521
```

3.3.9 Fragmentation Guard and Flood Defender

The PIX provides a built-in fragmentation guard and SYN flood defender. The Frag guard feature can cause problems if end systems attempt to negotiate a

TCP maximum segment size (mss) above 1380 bytes (PIX default) and will drop any fragmented packets, which may cause us some problems. The SYN flood feature will be enabled, but as we do not yet know enough about our environment, we will monitor the firewall very closely initially and may have to adjust the default threshold for the number of unanswered SYNs.

3.3.10 Object Groups

Object groups are a feature that allows hosts, protocols and services to be grouped. This simplifies the firewall configuration when defining NAT and access lists. For example, if one server called 'prodweb' is listening on TCP ports 21, 25 and 80, we can define an object group for those services. When we build the access list, rather than add a line for each protocol, we can just reference the object group.

We will use the following object groups for the GIAC implementation.

```
object-group network vpn_network_devices
  network-object 192.168.3.50 255.255.255.255
  network-object 192.168.3.51 255.255.255.255
  network-object 192.168.3.3 255.255.255.255
  network-object 192.168.3.4 255.255.255.255

object-group network part_sup_network_devices
  network-object 192.168.4.3 255.255.255.255
  network-object 192.168.4.4 255.255.255.255

object-group network web_network_devices
  network-object 192.168.5.3 255.255.255.255
  network-object 192.168.5.4 255.255.255.255
  network-object 192.168.5.23 255.255.255.255
  network-object 192.168.5.24 255.255.255.255
```

(Note: 192.168.5.24 and 192.168.5.24 are source group VIPs from the content switch. These CSS source groups Nat the source IP address of any syslog traffic from the real Ethernet switch addresses of 192.168.6.3 to 192.168.5.24 and 192.168.6.4 to 192.168.5.24)

```
object-group network external_network_devices
  network-object 10.64.8.1 255.255.255.255
 network-object 10.64.8.2 255.255.255.255
```

```
  network-object 223.223.223.5 255.255.255.255
  network-object 223.223.223.6 255.255.255.255

object-group network GIAC_vpn_pools
  network-object 10.10.1.0 255.255.255.0
  network-object 10.10.2.0 255.255.255.0
  network-object 10.10.3.0 255.255.255.0

object-group network internal_hosts
  network-object 172.25.100.0 255.255.255.0
  network-object 172.25.200.0 255.255.255.0

object-group network app_servers
  network-object 192.168.9.10 255.255.255.255
  network-object 192.168.9.11 255.255.255.255

object-group network spoofed_networks
  network-object 0.0.0.0 255.0.0.0
  network-object 1.0.0.0 255.0.0.0
  network-object 2.0.0.0 255.0.0.0
  network-object 10.0.0.0 255.0.0.0
  network-object 23.0.0.0 255.0.0.0
  network-object 31.0.0.0 255.0.0.0
  network-object 67.0.0.0 255.0.0.0
  network-object 68.0.0.0 252.0.0.0
  network-object 72.0.0.0 252.0.0.0
  network-object 80.0.0.0 240.0.0.0
  network-object 96.0.0.0 240.0.0.0
  network-object 112.0.0.0 252.0.0.0
  network-object 126.0.0.0 254.0.0.0
  network-object 127.0.0.0 255.0.0.0
  network-object 169.254.0.0 255.255.0.0
  network-object 172.16.0.0 255.240.0.0
  network-object 191.255.0.0 255.255.0.0
  network-object 192.168.0.0 255.255.0.0
  network-object 198.18.0.0 255.255.0.0
  network-object 201.0.0.0 255.0.0.0
  network-object 222.255.255.0 255.255.255.0
  network-object 224.0.0.0 224.0.0.0

object-group network ntp_servers
  network-object 223.223.223.23 255.255.255.255
  network-object 223.223.223.24 255.255.255.255

object-group network syslog_servers
  network-object 223.223.223.21 255.255.255.255
  network-object 223.223.223.22 255.255.255.255

object-group network isp_dns_servers
  network-object 202.139.83.3 255.255.255.255
  network-object 192.65.90.202 255.255.255.255

object-group service webservices tcp
  description inbound HTTP and SSL
  port-object eq www
port-object eq https

object-group service web_management tcp
  description web servers management
  port-object eq 8081
  port-object eq 9173
  port-object eq 16187

object-group network part_sup_vpn_pools
  network-object 10.10.10.0 255.255.255.0
  network-object 10.10.20.0 255.255.255.0
```

```
object-group network appservers
  network-object 192.168.9.10 255.255.255.255
  network-object 192.168.9.11 255.255.255.255

object-group service appserver_ports tcp
  description websphere ports
  port-object eq 9080
  port-object eq 9443

object-group network web_servers
  network-object 192.168.5.101 255.255.255.255
  network-object 192.168.5.102 255.255.255.255
  network-object 192.168.5.103 255.255.255.255
  network-object 192.168.5.104 255.255.255.255

object-group network inside_address_syslog_servers
  network-object 192.168.11.11 255.255.255.255
  network-object 192.168.11.12 255.255.255.255

object-group network syslog_servers_real
  network-object 192.168.11.11 255.255.255.255
  network-object 192.168.11.12 255.255.255.255

object-group network ntp_servers_real
  network-object 172.25.1.101 255.255.255.255
  network-object 172.25.1.102 255.255.255.255

object-group network inside_address_ntp_servers
  network-object 172.25.1.101 255.255.255.255
  network-object 172.25.2.102 255.255.255.255
```

3.3.11 Network Address Translation (NAT)

The PIX firewall requires that NAT is configured on the outside interface (even if NAT is not required, 'NO NAT" must be configured). For the GIAC implementation, NAT is required and will be used to translate outside, publicly routable IP addresses into an inside RFC 1918 IP address. The RFC1918 range will be used on the screened subnets and on the internal network. Outbound user traffic from the inside RFC1918 172.25.0.0 /16 ranges will be translated into the outside interface address of the PIX which is publicly routable address.

3.3.12 Translation Rules

The PIX uses a combination of translation rules and access lists to control traffic flowing through the firewall. A translation rule is a mandatory requirement to enable the PIX to permit traffic; without a translation rule on a given interface the PIX will drop any traffic on that interface.

By default, once a translation rule has been configured all traffic can traverse from a higher security level interface to a lower security level interface without the need to configure ACL's to permit any traffic. However the reverse is not true, so even if a translation rule has been configured, traffic cannot traverse from a lower level security interface to a higher level (i.e. outside to inside), unless an ACL has been defined permitting traffic.

Once translation rules have been defined, access lists are used to control the specific traffic to be permitted or denied.

A general rule of thumb for the PIX is to dynamic NAT for higher security to lower security translations (inside to outside) and static NAT for lower security to higher security translations (outside to inside).

Outside DMZ

First we need to define translation rules for the publicly accessible outside addresses into the internal hidden (RFC1918) range. This will be required for web, mail and DNS access from the Internet and for NTP, syslog and TFTP from the external routers and switches.

A summary of the required translations is shown in the following table:

Outside Address	Inside Address	Description
223.223.223.21	192.168.11.11	Syslog server 1
223.223.223.22	192.168.11.12	Syslog server 2
223.223.223.23	172.25.1.101	NTP server 1

Outside Address	Inside Address	Description
223.223.223.24	172.25.2.102	NTP server 2
223.223.223.50	192.168.5.12	External Mail Server
223.223.223.12	192.168.5.50	DNS Server
223.223.223.100	172.25.5.100	Web site VIP
223.223.223.13	172.25.11.13	TFTP Server

Table 2 – Outside address translations

Static NAT commands are used to define the required translation rules. To permit these inbound connections we configure:

```
Static (inside,outside) 223.223.223.21 192.168.11.11 netmask 255.255.255.255
Static (inside,outside) 223.223.223.22 192.168.11.12 netmask 255.255.255.255
Static (inside,outside) 223.223.223.23 172.25.1.101 netmask 255.255.255.255
Static (inside,outside) 223.223.223.24 172.25.2.102 netmask 255.255.255.255
Static (web,outside) 223.223.223.12 192.168.5.12 netmask 255.255.255.255
Static (web,outside) 223.223.223.50 192.168.5.50 netmask 255.255.255.255
Static (web,outside) 223.223.223.100 192.168.5.100 netmask 255.255.255.255
Static (inside,outside) 223.223.223.13 192.168.11.13 netmask 255.255.255.255 0 0
```

An access control list will be applied to the outside interface to control access for through the PIX from the outside DMZ.

VPN Screened Subnet

Next we define translation rules to enable VPN user's access to the various screened subnets and the internal network as required. The following translations are required:

Outside VPN address	Inside Address	Description
192.168.11.11	192.168.11.11	Syslog server 1
192.168.11.12	192.168.11.12	Syslog server 2
192.168.11.13	192.168.11.13	TFTP server
172.25.1.50	172.25.1.50	Internal DNS server
172.25.1.101	172.25.1.101	NTP server 1
172.25.2.102	172.25.2.102	NTP server 2
172.25.2.50	172.25.2.50	Internal Exchange server

Outside VPN address	Inside Address	Description
192.168.4.10	192.168.4.10	Partsup FTP server
192.168.4.11	192.168.4.11	Partsup Oracle server
223.223.223.100	192.168.5.100	Web Site VIP

Table 3 – PIX VPN interface address translations

Static NAT commands are used to define the required translation rules. To permit these inbound translations we configure:

```
static (inside,vpn) 192.168.11.11 192.168.11.11 netmask 255.255.255.255 0 0
static (inside,vpn) 192.168.11.12 192.168.11.12 netmask 255.255.255.255 0 0
static (inside,vpn) 192.168.11.13 192.168.11.13 netmask 255.255.255.255 0 0
static (inside,vpn) 172.25.1.50 172.25.1.50 netmask 255.255.255.255 0 0
static (inside,vpn) 172.25.1.101 172.25.1.101 netmask 255.255.255.255 0 0
static (inside,vpn) 172.25.2.102 172.25.2.102 netmask 255.255.255.255 0 0
static (inside,vpn) 172.25.2.50 172.25.2.50 netmask 255.255.255.255 0 0
static (partsup,vpn) 192.168.4.10 192.168.4.10 netmask 255.255.255.255 0 0
static (partsup,vpn) 192.168.4.11 192.168.4.11 netmask 255.255.255.255 0 0
```

Note we configure the source and NAT address the same. We use a 'static' NAT statement to create the required translation rule, but maintain the original IP addresses.

External users (i.e. from the Internet) will be returned the IP Address 223.223.223.100 for the GIAC web site. However, we need to provide access to the GIAC web site for GIAC VPN users. To achieve this we will create an additional translation rule that allows the outside IP address assigned to the web site (223.223.223.100) to also be used on the vpn interface. Cisco refers to this as bi-directional NAT.

The following is already configured on the outside interface.

```
static (web,outside) 223.223.223.100 192.168.5.100 netmask 255.255.255.255 0 0
```

We add this on the vpn interface to set up the bi-directional NAT.

```
static (web,vpn) 223.223.223.100 192.168.5.100 netmask 255.255.255.255 0 0
```

An access list will be applied to the vpn interface to control access through the PIX from the vpn screened subnet.

We need to enable our VPN users to access the Internet. Whilst this imposes some overhead for VPN users, in that they have to first come across the Internet through their VPN connection to access the Internet, this is a policy that GIAC have set. Split tunnelling over VPN connections is not permitted. Internet access via this method is only enabled for the GIAC remote VPN users. We do not allow suppliers and partners to utilise our Internet link for any Internet connections, other than the VPN connection to GIAC. Note that we are using a different IP address as the external PAT address for VPN users. To permit these translations from the vpn interface we configure:

```
Global (outside) 2 223.223.223.200
nat (vpn) 2 10.10.1.0 255.255.255.0 0 0
nat (vpn) 2 10.10.2.0 255.255.255.0 0 0
nat (vpn) 2 10.10.3.0 255.255.255.0 0 0
```

Web screened Subnet

Next we define the translation rules for the web servers to access the Websphere application servers and the LDAP proxy server and for the network devices to access the syslog, NTP and TFTP servers. The following translations are required:

Outside Web address	Inside Address	Description
192.168.11.11	192.168.11.11	Syslog server 1
192.168.11.12	192.168.11.12	Syslog server 2
192.168.11.13	192.168.11.13	TFTP server
172.25.1.101	172.25.1.101	NTP server 1
172.25.2.102	172.25.2.102	NTP server 2
172.25.2.50	172.25.2.50	Internal Exchange server
192.168.9.100	192.168.9.100	LDAP Proxy
192.168.9.10	192.168.9.10	WebSphere Application Server 1
192.168.9.11	192.168.9.11	WebSphere Application Server 2

Table 4 – PIX WEB interface address translations

To permit these inbound translations we configure:

```
static (inside,web) 192.168.11.11 192.168.11.11 netmask 255.255.255.255 0 0
static (inside,web) 192.168.11.12 192.168.11.12 netmask 255.255.255.255 0 0
static (inside,web) 192.168.11.13 192.168.11.13 netmask 255.255.255.255 0 0
static (inside,web) 172.25.1.101 172.25.1.101 netmask 255.255.255.255 0 0
static (inside,web) 172.25.2.102 172.25.2.102 netmask 255.255.255.255 0 0
static (inside,web) 172.25.2.50 172.25.2.50 netmask 255.255.255.255 0 0
static (inside,web) 192.168.9.100 192.168.9.100 netmask 255.255.255.255 0 0
static (inside,web) 192.168.9.10 192.168.9.10 netmask 255.255.255.255 0 0
static (inside,web) 192.168.9.11 192.168.9.11 netmask 255.255.255.255 0 0
```

(Note we configure the source and NAT address to be the same again, as we do not want to actually change any addresses, but still have to create the translation rule)

We need to add some dynamic PAT to enable the DNS and Mail servers to make outbound connections. For this we configure:

```
global (outside) 1 interface
nat (web) 1 192.168.5.12 255.255.255.255 0 0
nat (web) 1 192.168.5.50 255.255.255.255 0 0
```

This will perform NAT on all outbound connections from the DNS and external Mail server to the IP address of the PIX outside interface.

An access control list will be applied to the web interface to control access for through the PIX from the web screened subnet.

Partsup screened Subnet

Now we define the translations required for the partsup screened subnet. We need to have translations for the network devices to access the syslog, NTP and TFTP servers. There should be no outbound connections initiated from the Oracle or FTP servers.

The following translations are required:

Outside Web address	Inside Address	Description
192.168.11.11	192.168.11.11	Syslog server 1
192.168.11.12	192.168.11.12	Syslog server 2
192.168.11.13	192.168.11.13	TFTP server
172.25.1.101	172.25.1.101	NTP server 1
172.25.2.102	172.25.2.102	NTP server 2

Table 5 – PIX Partsup interface address translations

To enable these translations we configure:

```
static (inside,partsup) 192.168.11.11 192.168.11.11 netmask 255.255.255.255 0 0
static (inside,partsup) 192.168.11.12 192.168.11.12 netmask 255.255.255.255 0 0
static (inside,partsup) 192.168.11.13 192.168.11.13 netmask 255.255.255.255 0 0
static (inside,partsup) 172.25.1.101 172.25.1.101 netmask 255.255.255.255 0 0
static (inside,partsup) 172.25.2.102 172.25.2.102 netmask 255.255.255.255 0 0
```

Again we have configured the source and NAT address to be the same as we do not want to actually NAT any IP addresses. An access control list will be applied to the partsup interface to control access for through the PIX from the network devices on the partsup screened subnet.

Inside Interface

Next we define dynamic NAT for the internal users accessing the Internet. We will actually use port address translation (PAT), which will translate all internal IP addresses to a single outside IP address. We will use the IP address of the firewalls outside interface for all PAT. Here the 'nat (inside) command is followed by a '1'. This is a reference to the PAT pool defined in the 'global' statement immediately above it, which in this case is configured to use the outside interface IP Address. Rather than 'interface', this could have optionally specified an IP address pool. The '172.25.100.0 255.255.255.0 and 172.25.200.0 255.255.255.0' statements are for the range of user IP address on the inside that will be matched against this dynamic PAT statement. The '192.168.9.50 255.255.255.255' address is to match the proxy server, as all internal user's web and ssl sessions will be via the proxy.

```
global (outside) 1 interface
nat (inside) 1 192.168.9.50 255.255.255.255 0 0
nat (inside) 1 172.25.100.0 255.255.255.0 0 0
nat (inside) 1 172.25.200.0 255.255.255.0 0 0
```

Note that we are not using any static NAT for outbound access.
A peculiarity and sometimes confusing feature of the PIX is that we still have to define a 'NAT' statement to create a translation rule, even if we do not want to actually translate any IP addresses. Remember, the PIX must have a translation rule for a given IP address, before it will pass traffic.

There is such a requirement for the following inside addresses:

- Administration access from the network management station to network devices on the vpn, partsup and web screened subnets.
- Internal user access to the web servers for management
- Internal user access to the Content Transformation Engine.
- Internal user access to the web site VIP.
- Internal user access to the partsup FTP server and partsup Oracle server
- Internal DNS server to external DNS server

For the other interfaces we have used static NAT statements to set up the translation and maintain the same IP addressing as the access has been from a lower security interface to a higher security interface.

For outbound traffic from the inside interface however, we will use dynamic NAT, as this is a higher security interface traversing to a lower security interface and will reference an ACL for IP addresses that should not have NAT applied. This is configured with the following statement:

nat (inside) 0 access-list inside_outbound_nat0_acl

The '0' after the (inside) means 'NO NAT', so any IP addresses that match the access list will NOT have their addresses translated.

An access control list will be applied to the inside interface to control access for through the PIX from the inside network.

It should be noted that the internal users will not be aware of any of the 192.168.x.x address ranges in the security zone. All internal users will connect to a 172.25.10.0 /24 IP address which the Checkpoint firewall will then perform destination NAT to the 192.168.x.x range.

A table of Checkpoint firewall translation is shown below:

Original Destination Address	Translated By Checkpoint Firewall to Destination IP address	Description
172.25.10.11	192.168.11.11	syslog server 1
172.25.10.12	192.168.11.12	syslog server 2
172.25.10.50	192.168.10.50	Prod Oracle Server
172.25.10.51	192.168.4.10	Replica Oracle Server
172.25.10.52	192.168.4.11	supplier ftp server
172.25.10.53	192.168.9.50	web proxy server
172.25.10.55	192.168.5.50	External DNS server
172.25.10.60	192.168.10.60	Prod LDAP Server
172.25.10.70	192.168.5.12	External Mail Server
172.25.10.81	192.168.9.10	WebSphere 1
172.25.10.82	192.168.9.11	WebSphere 2
172.25.10.101	192.168.5.101	web server 1
172.25.10.102	192.168.5.102	web server 2
172.25.10.103	192.168.5.103	web server 3
172.25.10.104	192.168.5.104	web server 4
172.25.10.111	192.168.5.111	Content Transformation Engine

Table 6 - Checkpint Firewall Translations

3.3.13 DNS Doctoring

Internal users need access to the GIAC web site, so we will use the DNS doctoring feature of the PIX to modify the DNS reply packet for queries from internal users.

The external DNS server will respond to DNS queries for www.giac.com.au with the DNS A-record of 223.223.223.100. The PIX will modify the DNS reply packet and change the 223.223.223.100 to 172.25.10.100, which is within the internal Checkpoint firewall inside interface range. As we do not allow the internal users direct access to any of 192.168.0.0 ranges, internal users will then connect to www.giac.com.au as 172.25.10.100 and the checkpoint will perform NAT to the 192.168.5.100 VIP address of the content switch.

On the PIX we have previously defined a ‘NO NAT’ rule for the 192.168.5.100 address from the internal users, so the PIX will just pass this through to the VIP address on the content switch.
The DNS doctoring feature is configured on the PIX as follows:

```
static (inside,outside) 223.223.223.100 172.25.10.100 dns netmask 255.255.255.255 0 0
```

3.3.14 Access Lists

The translation rules have been defined above; we now need to control traffic against these translation rules with the use of access control lists. We will extensive use of the object groups that we defined earlier in our access lists. Another advantage of using object groups, is that in order to change the access list, we simply need t o change the object group, rather than the access list. With object groups, specific lines with the object group can deleted, rather than the entire object group or access list, which makes changes easier.

Outside Access In

We need define the traffic that we will allow into GIAC from the Internet. This will be a similar access list to the ingress filter on the external routers. Lines preceded by ‘#’ are comments. As with a router, the PIX will act on the first match, so ACL rule ordering is important.

```
#Deny any spoofed source IP address
access-list outside_access_in deny ip object-group spoofed_networks any

# Permit traffic to web site services
access-list outside_access_in permit tcp any host 223.223.223.100 object-
group webservices

#Permit access to DNS server – queries only
access-list outside_access_in permit udp any host 223.223.223.50 eq dns
```

```
#permit access to external mail server
access-list outside_access_in permit tcp any host 223.223.223.12 eq smtp

# permit syslog from external dmz devices
access-list outside_access_in permit udp object-group
external_network_devices object-group syslog_servers eq syslog

#permit certain ICMP traffic
access-list outside_access_in permit icmp any 223.223.223.0 255.255.255.0
echo-reply
access-list outside_access_in permit icmp any 223.223.223.0 255.255.255.0
unreachable
access-list outside_access_in permit icmp any 223.223.223.0 255.255.255.0
time-exceeded

# permit ntp from external dmz devices
access-list outside_access_in permit tcp object-group
external_network_devices object-group ntp_servers eq 123

#Permit dns zone transfers only to the ISP dns servers
access-list outside_access_in permit tcp object-group isp_dns_servers host
223.223.223.50 eq domain

#Permit TFTP from the network devices to the TFTP server
access-list outside_access_in permit udp object-group
external_network_devices host 223.223.223.13 eq tftp

# Deny everything Else
access-list outside_access_in deny ip any any
```

Note: For all other interfaces on the PIX we do not need to specify the anti spoofing object group in our access list as the 'verify reverse path' feature of the PIX will drop any spoofed source IP addresses on any of the other interfaces. The anti-spoofing object group is required on the outside interface as the PIX firewall has a default route out of this interface.

VPN Access In

Next we define traffic that we will permit in from the VPN connections and vpn segment. Again we reference the previously defined object groups to simplify the ACL's.

```
#Permit GIAC VPN users access to the GIAC web site and Internet for HTTP
and SSL
access-list vpn_access_in permit tcp object-group GIAC_vpn_pools any object-
group webservices

#Permit GIAC VPN users access to the internal DNS server
access-list vpn_access_in permit udp object-group GIAC_vpn_pools host
172.25.1.50 eq domain
```

```
#Permit syslog for the vpn network devices
access-list vpn_access_in permit udp object-group vpn_network_devices
object-group inside_address_syslog_servers eq syslog

#Permit GIAC VPN users access to the internal Exchange server
access-list vpn_access_in permit tcp object-group GIAC_vpn_pools host
172.25.2.50 eq smtp

#Permit partner and supplier VPN pools access to the Oracle and FTP servers
on part_sub screened subnet
access-list vpn_access_in permit tcp object-group part_sup_vpn_pools host
192.168.4.10 eq sqlnet
access-list vpn_access_in permit tcp object-group part_sup_vpn_pools host
192.168.4.11 eq ftp

#Permit GIAC fotune cookie processing vpn users access to the
part_sub_screened subnet
access-list vpn_access_in permit tcp 10.10.2.0 255.255.255.0 host
192.168.4.10 eq sqlnet
access-list vpn_access_in permit tcp 10.10.2.0 255.255.255.0 host
192.168.4.11 eq ftp

#Permit GIAC admin staff vpn users access to the part_sub_screened subnet
access-list vpn_access_in permit tcp 10.10.3.0 255.255.255.0 host
192.168.4.10
access-list vpn_access_in permit tcp 10.10.3.0 255.255.255.0 host
192.168.4.11

#Permit ntp for the vpn network devices
access-list vpn_access_in permit tcp object-group vpn_network_devices
object-group inside_address_ntp_servers eq 123

#Permit the vpn network devices access to the TFTP server.
access-list vpn_access_in permit udp object-group vpn_network_devices host
192.168.11.13 eq tftp

#Deny everything else
access-list vpn_access_in deny ip any any
```

Web Access In

Next we define traffic that we will permit in from the web segment.

```
# Permit the web serves to talk to the Websphere application servers – Note:
all outbound web server traffic is source nated to 192.168.5.200 as the source
ip address by the content switch before it gets to the PIX.
access-list web_access_in permit tcp host 192.168.5.200 object-group
app_servers object-group appserver_ports

# Permit web servers access to the LDAP proxy
```

```
access-list web_access_in permit tcp host 192.168.5.200 host 192.168.9.100
eq ldap

#Permit syslog for the web network devices
access-list web_access_in permit udp object-group web_network_devices
object-group inside_address_syslog_servers eq syslog

#Permit outbound DNS queries from the external DNS server
access-list web_access_in permit udp host 192.168.5.50 any eq domain

#Permit outbound SMTP connections from the external mail server
access-list web_access_in permit tcp host 192.168.5.12 any eq smtp

#Permit DNS zone transfers to the ISP DNS servers
access-list web_access_in permit tcp host 192.168.5.50 object-group
isp_dns_servers eq domain

#Permit ntp for the web network devices
access-list web_access_in permit tcp object-group web_network_devices
object-group inside_address_ntp_servers eq 123
#Permit the web network devices access to the TFTP server.
access-list web_access_in permit udp object-group web_network_devices host
192.168.11.13 eq tftp

#Deny everything else
access-list web_access_in deny ip any any
```

Partsup Access In

Next we define traffic that we will permit in from the partsup segment. The partsup screened subnet will not have any connections initiating from it, except from the network devices.

```
#Enable syslog for the part_sup network devices
access-list partsup_access_in permit udp object-group
part_sup_network_devices object-group inside_address_syslog_servers eq
syslog

#Enable ntp for the part_sup network devices
access-list partsup_access_in permit tcp object-group
part_sup_network_devices object-group inside_address_ntp_servers eq 123

#Permit the partsup network devices access to the TFTP server.
access-list partsup_access_in permit udp object-group
part_sup_network_devices host 192.168.11.13 eq tftp

#Deny everything else
access-list part_sup_access_in deny ip any any
```

Sync Access In

The sync interface will not be permitted to accept any traffic. All traffic will be explicitly denied.

```
access-list sync_access_in deny ip any any
```

Internal Access In

Now we need to define an ACL to restrict the permitted traffic from internal users.

```
# Allow HTTP, SSL access out from the proxy server only
access-list inside_access_in permit tcp host 192.168.9.50 any object-group
webservices

#Permit recursive DNS queries from the internal DNS server to the external
DNS server
access-list inside_access_in permit udp host 172.25.1.50 host 192.168.5.50 eq
domain

#Permit outbound FTP from the proxy server only
access-list inside_access_in permit tcp host 192.168.9.50 any eq ftp

#Permit LDAP from internal users – used for PGP key server connection
access-list inside_access_in permit tcp object-group internal_hosts any eq ldap

#Permit POP3
access-list inside_access_in permit tcp object-group internal_hosts any eq
pop3

#Permit news
access-list inside_access_in permit tcp object-group internal_hosts any eq nntp

#Permit internal Echange server to external mail server
access-list inside_access_in permit tcp host 172.25.2.50 host 192.168.5.12 eq
smtp

#Allow internal hosts access to the replicae oracle server
access-list inside_access_in permit tcp object-group internal_hosts host
192.168.4.10 eq sqlnet

#Allow internal hosts access to the web servers for content management
access-list inside_access_in permit tcp object-group internal_hosts object-
group web_servers object-group web_management

#Allow internal hosts access to the content transformation engine for
management
access-list inside_access_in permit tcp object-group internal_hosts host
192.168.5.111 eq 9001

#Allow Telnet from management station
access-list inside_access_in permit tcp host 192.168.11.50 any eq telnet
```

```
#Deny any other Telnet connections to the GIAC network devices
access-list inside_access_in deny tcp object-group internal_hosts object-group
external_network_devices eq telnet
access-list inside_access_in deny tcp object-group internal_hosts object-group
vpn_network_devices eq telnet
access-list inside_access_in deny tcp object-group internal_hosts object-group
web_network_devices eq telnet
access-list inside_access_in deny tcp object-group internal_hosts object-group
part_sup_network_devices eq telnet

#Permit all other outbound Telnet connections
access-list inside_access_in permit tcp object-group internal_hosts any eq
telnet

#Deny certain ICMP traffic as ths could be used to map the GIAC network.
access-list inside_access_in deny icmp any any time-exceeded
access-list inside_access_in deny icmp any any unreachable
access-list inside_access_in deny icmp any any echo-reply

#Permit all other ICMP traffic
access-list inside_access_in permit icmp object-group internal_hosts any

#Permit ICMP traffic from the management station
access-list inside_access_in permit icmp host 192.168.11.50 any
#Deny everything else
access-list inside_access_in deny ip any any
```

Finally we need to define network management traffic that we will allow through the PIX without NAT. For access from the management station to the vpn, web and partsup segments and for internal users accessing the partsup and web segments we will not be translating any addresses. This access list is referenced by the statement:

‘*nat (inside) 0 access-list inside_inbound_nat0_acl*’

IP addresses that match this ACL will bypass NAT. Note that we again reference the previously defined object groups to build our ACL’s. This simplifies the ACL construction.

```
# NO NAT management host to vpn segment network devices
access-list inside_outbound_nat0_acl permit ip host 192.168.11.50 object-
group vpn_network_devices

# NO NAT management host to part_sup segment network devices
access-list inside_outbound_nat0_acl permit ip host 192.168.11.50 object-
group part_sup_network_devices

# NO NAT management host to web segment network devices
```

```
access-list inside_outbound_nat0_acl permit ip host 192.168.11.50 object-
group web_network_devices

# NO NAT internal addresses to the partsup Oracle server
access-list inside_outbound_nat0_acl permit ip object-group internal_hosts
host 192.168.4.10

# NO NAT internal addresses to the partsup FTP server
access-list inside_outbound_nat0_acl permit ip object-group internal_hosts
host 192.168.4.11

# NO NAT internal addresses to the web servers for content management
access-list inside_outbound_nat0_acl permit ip object-group internal_hosts
object-group web_servers

# NO NAT internal addresses to the content transformation engine for
management. Note 192.168.5.111 is the VIP address of the CSS. This CSS
performs NAT on this to the real CTE address of 192.168.6.111.
access-list inside_outbound_nat0_acl permit ip object-group internal_hosts
host 192.168.5.111

#NO NAT the internal DNS server to the external DNS server
access-list inside_outbound_nat0_acl permit ip host 172.25.1.50 host
192.168.5.50
```

Now that we have defined the access lists, we need to apply then to the relevant interfaces. We will apply all of the access lists as inbound access lists.

```
access-group web_access_in in interface web
access-group vpn_access_in in interface vpn
access-group sync_access_in in interface sync
access-group part_sup_access_in in interface partsup
access-group outside_access_in in interface outside
access-group inside_access_in in interface inside
```

We have ordered our access lists in what we think is the vest sequence, but we will review these after a period of time and check the hit rate on each ACL line and re-order the object groups and ACL's as required.

This completes the PIX tutorial.

3.4 VPN

3.4.1 Overview

VPN services are provided via a Cisco 3015 VPN concentrator unit. This has two interfaces connected to the network; one interface connected to the external DMZ to terminate the inbound VPN connections and one interface connected to the secure vpn screened subnet to initiate the internal communication. This interface is dedicated to the vpn devices; there are no hosts located on this screened subnet.

Inbound VPN connections connect via IPSEC to the external or 'public' interface of the concentrator. IPSEC connections are terminated on the concentrator and clients are allocated a VPN IP address from the pools 10.10.1.0 /24, 10.10.2.0 /24, 10.10.3.0 /24, 10.10.10.0 /24 and 10.10.20.0 /24. Client traffic is then passed as 'clear text' out of the internal or 'private' interface of the concentrator and inspected by the PIX.

For the site to site connections we will be using pre shared keys as at this stage GIAC partners and suppliers, for various reasons, are not willing or capable of moving to digital certificates.

For GIAC remote users however, a PKI solution will be used to provide a secure VPN with two factor authentication. This uses Verisign digital certificates for authentication which are stored by each remote vpn user on USB tokens.

Verisign will host the GIAC private certificate authority (CA). This means that the root certificate will be self signed by GIAC and will not be signed by Verisign. An identity certificate will be generated for the VPN concentrator and a user certificate will be generated for each user. Additionally there will be an administrator's certificate.

Each remote access VPN user will be equipped with an Aladdin USB eToken which will store the user certificate. This requires that the Aladdin Run Time Environment (RTE) software be installed on each end-user machine.

The Cisco VPN client software will be installed on each user machine. One of the reasons that the Cisco client has been chosen over the built in Windows 2000 IPSEC client is that the Cisco client provides firewall functionality for the client machine.

Both the Cisco VPN client software and the Aladdin RTE software have been incorporated into the GIAC laptop standard operating environment (SOE) build.

3.4.2 VPN software

GIAC will be deploying a Cisco VPN 3015 concentrator running software version 3.6 – vpn3000-3.6.7.F-9.bin. The client software used will be the Cisco VPN Client version 3.6.4.A.

3.4.3 Client Requirements

All VPN users need to install the USB token drivers, token utility software and the Cisco VPN client software on their workstations. This has been completed as part of the SOE build.

3.4.4 PKI Certificates

Two types of certificates are used; *user certificates*, which are stored by each user on their USB token and an *identify certificate* which is stored on the VPN concentrator. All of these certificates are signed by the Certificate Authority (CA), which in this case is GIAC. This is an important point; the certificates are self-signed by GIAC, and NOT by Verisign. Verisign is responsible for setting up and hosting the private CA, but GIAC has full control over the certificate management.

Web access is provided to the CA site hosted by Verisign which enables users to request certificates and administrators to manage certificates; process user certificate requests, revoke certificates etc. Administrator access is controlled by a special administrator certificate. For security, this will be stored on a dedicated USB token that will be locked in a secure cabinet. A signing out procedure must be followed to obtain this administrator USB token.

3.4.5 VPN Mode

Tunnel mode only will be permitted by users connected to VPN; split tunnel mode will be explicitly prohibited by the setting of group policies on the concentrator. These are pushed down to the VPN client when the client connects to the VPN.

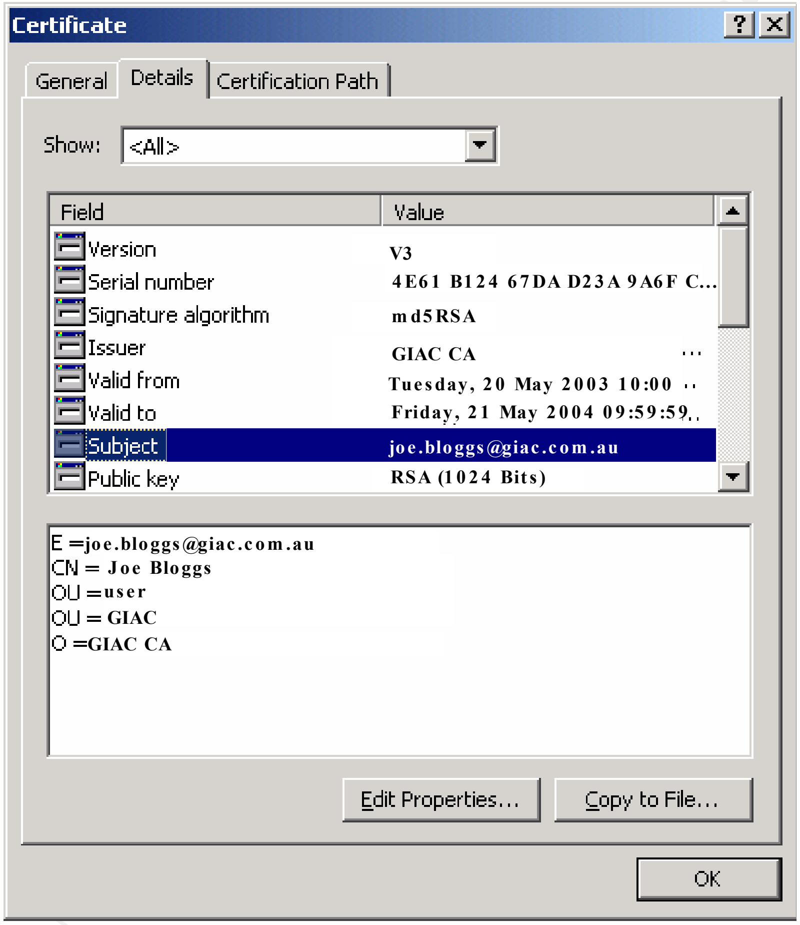
3.4.6 VPN User Groups

We will be using the following user groups for GIAC staff:

- General GIAC users
- GIAC Fortune Cookie Processing staff
- GIAC Administrators

These groups will be assigned to different IP address pools with different access rights, depending on the group. The firewall polices will match the access rights assigned to each address pool.

The group assignment is controlled by the 'OU' field in the user certificate. We will have an OU of 'GIAC' and then sub OU's for each user group. The screenshot below shows a sample certificate.



Show: <All>

Field	Value
Version	V3
Serial number	4E61 B124 67DA D23A 9A6F C...
Signature algorithm	md5RSA
Issuer	GIAC CA ...
Valid from	Tuesday, 20 May 2003 10:00 ..
Valid to	Friday, 21 May 2004 09:59:59..
Subject	joe.bloggs@giac.com.au
Public key	RSA (1024 Bits)

E =joe.bloggs@giac.com.au
CN = Joe Bloggs
OU =user
OU = GIAC
O =GIAC CA

Diagram 2 – Certificate Screenshot

Site to Site (or LAN to LAN) VPN connections will be configured for the partner and supplier VPN connections. We will use PGP to securely distribute the keys to the partner and supplier administrators and they will be responsible for configuring their VPN devices.

Initially the partners and suppliers will require the same level of access, but we are using different groups to make any future changes easier. This aids us in identifying the user groups for any logging.

3.4.7 IPSEC Overview

IPSEC is actually a framework consisting of multiple protocols including:

- IKE – Internet Key Exchange. This is used to define and negotiate security assignments (SA's), key generation and authentication data.
- AH – Authentication Header. This adds authentication information to the header of each packet and the receiving end can verify that the packet has not been altered. This provides packet integrity but does not provide any encryption (confidentiality). AH cannot be used with NAT.
- ESP – Encapsulating Security Payload. This provides encryption (confidentiality) by encrypting the data payload of the packet, but does not provide any authentication of the packet headers. ESP can operate in tunnel mode which inserts the IP datagram into an encrypted portion of the ESP payload or transport mode where the ESP header is inserted into another IP datagram, which can be TCP or UDP.
- Both ESP and AH use sequence numbers which provides protection against replay attacks.

3.4.8 GIAC IPSEC Configuration

As one of our partners uses NAT and it is probable that any future partners and suppliers will use NAT, we cannot use AH, so we have to use tunnel mode ESP. For the GIAC staff VPN users we will use transport mode ESP over TCP, using TCP port 10000.

GIAC will use ESP/SHA/HMAC as the ESP authentication mechanism. This uses Hashed Message Authentication Coding (HMAC) using the SHA1 hash function, which uses a 160 bit key. This has more processing overhead than the MD5 hash function (which uses a 128 bit key), but is more secure.

For encryption we have set a policy of using 3DES as the encryption standard. Luckily all of our partners and suppliers can all support 3DES. Our strategy is to move towards AES in the future, but we will have to review this as required for any future partners and suppliers.

AES is the Advanced Encryption Standard which uses an algorithm called Rijndeal developed by Joan Daemen and Vincent Rijmen. AES is seen as the replacement for triple DES (3DES). More information can be found at:

- http://www.esat.kuleuven.ac.be/~rijmen/rijndael/

Note the CRL checking will be performed via HTTP, not LDAP, se we need to ensue that we permit HTTP requests from the VPN concentrator to the Verisign site.

3.4.9 VPN Base Concentrator Configuration

To configure the concentrator to use certificates, first enter the basic concentrator configuration by attaching to the concentrator through the console port and a terminal emulator.

Enter the basic details below:

Item	Setting	Description
Time	Current Time	
Date	Current Date	
Timezone	+9	Sydney local timezone
Ip addresses	223.223.223.75	Outside (public) address
Ip address	192.168.3.50	Inside (private) address
Speed	100 Mbps	
Duplex	Full	
System name	vpn01	
DNS server	192.168.5.50	External DNS server
Domain	www.giac.com.au	Dns domain name
Default gateway	192.168.3.1	Interface of PIX vpn DMZ
PPTP	disabed	Not required, so disabled
L2TP	disabled	Not required, so disabled
IPSEC	Enabled	We use IPSEc for our vpn access
Client specified IP address assignment	Disabled	We do nto allow client to set the vpn IP address
Per user address assignment	Enable	We will allocate an IP address for each user.
DHCP	Enable	Use DHCP server on concentrator to manage address allocations
Configured pool address assignment	Disable	We will configure this from the browser later
Authentication	Continue	We will configure this from the browser later
IPSEC group name	GIAC	

Table 7 – Base VPN Concentrator Configuration

Save and exit from the initial configuration screen. Then we need to attach the concentrator to the network and connect via a browser to the concentrator's internal (private) IP address.

3.4.10 VPN Concentrator Device Certificate Configuration

To install the IPSEC device certificate, perform the following.

1) Save the CA certificate to a file in X.509 format. To download the CA certificate you must log into the CA and download the relevant file. This is typically a .tar file, but the process and file types may differ depending on which CA is being used. You must save or copy this file to the machine that you will be using to configure the concentrator.
2) Import the CA into the concentrator from the "Administration->Certificate Management->Install CA Certificate" menu and select "Upload File from Workstation" to install the certificate on the concentrator.

(Depending which CA you are using, you may have to select from several files, but typically you need the file with the *".x509"* extension. Check with you CA provider if you are unsure.)

Once the CA has been installed you need to create an identity certificate. This is performed as follows:

3) From the "certificate management" menu item, select the "Enrolment" link. Select "identity certificate" for the type.
4) Select Click "Enroll via PKCS10 Request (Manual)" for the method and enter values to identify the VPN 3015. We will set the key size to RSA 1024 bits.
5) The select "Enrol". This will generate a certificate signing request (CSR). You then need to save this to a text file.

Then you need to connect to the CA hosting site and select the IPSEC enrolment option (Again this process may differ between hosting CA's).

You will then be required to enter several pieces of information, including your email address and will have to either upload or paste the CSR into the browser window, on the CSR enrolment page. You then need to click on the 'SUBMIT' option to complete the enrolment request.

You (or the approved administrator) then need to log into the CA administration page (using the administration user certificate) and approve the request.

Once approved, an email containing the certificate will be sent to the email address specified on the enrolment page. See example below:

```
-----BEGIN CERTIFICATE-----
MIIDKjCCApOgAwIBAgIQLDfnMyRO7ptqbfyNhcviCzANBgkqhkiG9w0BAQQFADCB
nTEXMBUGA1UEChMORGltZW5zaW9uIERhdGExIjAgBgNVBAsTGUZvciBUZXN0aW5n
```

```
IFB1cnBvc2VzIE9ubHkxPzA9BgNVBAsTNlRlcm1zIG9mIHVzZSBhdCBodHRwOi8v
d3d3LmVzaWduLmNvbS5hdS90ZXN0Q2xpZW50Q1BTLzEdMBsGA1UEAxMUVlBOIGlu
IGEgYm94IFRlc3QgQ0EwHhcNMDIwNzE5MDAwMDAwWhcNMDMwNzE5MjM1OTU5WjBh
MQ4wDAYDVQQDEwV2cG4wMTEMMAoGA1UECxMDSVRTMRMwEQYDVQQKEwpXaWxzb24g
SFRNMREwDwYDVQQHEwhCcmlzYmFuZTEMMAoGA1UECBMDUUxEMQswCQYDVQQGEwJB
VTCBnTANBgkqhkiG9w0BAQEFAAOBiwAwgYcCgYEAhvCx9PawHvDpZufN/5yZtkV9
xPtrWLfbr9BcYBbO6PqQrsFZ6FewTAJvYVrre5zNJwwnIWQ+0Pw2fL6H2arj9P7K
PlcW7Ci3HR6xYPJl/eH4BDFqBpRw/m1jT8Ux4JQnyzbzgnn6IrshwTJZQGw0f2xi
yon5FNDQjiNuhskeal0CAQWjgacwgaQwIQYDVR0RBBowGIIWdnBuMDEud2lsc29u
aHRtLmNvbS5hdTAJBgNVHRMEAjAAMGcGA1UdHwRgMF4wXKBaoFiGVmh0dHA6Ly9v
bnNpdGVjcmwtdGVzdC5lc2lnbi5jb20uYXUvRGltZW5zaW9uRGF0YVZQTklQU2Vj
dGVzdGFjY291bnQvTGF0ZXN0Q1JMSVBTZWMuY3JsMAsGA1UdDwQEAwIFoDANBgkq
hkiG9w0BAQQFAAOBgQBmMoQLVfnzZYCko9VFJf9o/rvw9uk6E1QVWA0241PuYNOn
6CvAWmtu/R2xLyzNvnNxKGG+65dqMt2tiAFMoJQpuvvlazxzgOPIh6rWp0EAxBtZ
CBWyX6wNGdxGFPvY3mW+O1cUL5lm8h60usQXiccEXcegDlVQqS4+bRx0YWcVQg==
-----END CERTIFICATE-----
```

You need to copy the certificate to a text file and log back into the concentrator. Go to the *“Administration->Certificate <Management-> Install certificate obtained via Enrolment”* menu option and select *“install”*
Paste the certificate into the text box in the browser window.
Navigate back to the *“Administration>Certificate management”* screen and you should now see both the CA certificate and the Identity certificate.

The next step is to complete the remainder of the concentrator configuration. The following details need to be completed.

Configuration | System | Tunneling Protocols | IPSec | IKE Proposals	Added an Active Proposal “GIACVPN”, and made it active. Authentication Mode: RSA Digital Certificate Authentication Algorithm: MD5/HMAC-128 Encryption Algorithm: 3DES-168 Diffie-Hellman Group: Group 2 (1024-bits) Lifetime Measurement: Time Data Lifetime: 10000 Time Lifetime: 86400
Configuration | System | Tunneling Protocols | IPSec | IPSec over TCP	Enabled. Port 10000
Configuration | Policy Management | Traffic Management | Security Associations	Modified “ESP-3DES-MD5” Inheritance: From Rule IPSec Parameters Authentication Algorithm: ESP/MD5/HMAC-128 Encryption Algorithm: 3DES-168 Encapsulation Mode: Tunnel Perfect Forward Secrecy: Disabled

	Lifetime Measurement: Time Data Lifetime: 10000 Time Lifetime: 28800 IKE Parameters IKE Peer: 0.0.0.0 Negotiation Mode: Main Digital Certificate: GIACCERT Certificate Transmission: Identify certificate only IKE Proposal: GIACVPN
Configuration | User Management | Groups	Added Groups: partners, suppliers, giac, admin, cookies Identity Tab Group Name: Private Client Onsite Password: <doesn't matter, but must be entered> Type: Internal General Tab Filter: None – Inherit selected Tunneling Protocols: IPSec- no inherit Other setting left as default IPSec Tab IPSec SA: ESP-3DES-MD5- no inherit IKE Peer Identity Validation: Required – no inherit IKE Keepalives – selected – inherit selected Tunnel Type: Remote Access – no inherit Authentication: None <important setting> Mode Configuration – Selected – no inherit
Configuration | User Management | Groups | Address Pools	Added the following pools 10.10.10.1 – 10.10.10.254 10.10.20.1 – 10.10.20.254 10.10.1.1 – 10.10.1.254 10.10.2.1 – 10.10.2.254 10.10.3.1 – 10.10.3.254
Administration | Certificate Management | Configure CA Certificate	CRL Retrieval Policy – select 'Use static CRL distribution points' In the Static CRL Distribution Points box

	enter : ‘http://**hostingCA.com/**GIAC/LatestCRL.crl’ Certificate Acceptance Policy – Accept Subordinate CA Certificates: Selected Accept Identity Certificates signed by this user

Table 8 – Additional VPN Concentrator Configuration

As the VPN concentrator is not protected by a firewall for inbound connections, access lists will be configured on the concentrator to allow only the required protocols in on the ‘public’ interface.

4. Verify the Firewall Policy

Before we can commence our firewall audit we need to define a plan of exactly what we want to test and what we expect to get as results.

4.1 Plan the Audit

To provide the most flexibility, to enable address spoofing to be easily tested and to simulate a host takeover, we have decided to attach a laptop directly to each of the firewall interfaces in turn. In this scenario we are also assuming that the external routers have been compromised and all filtering on the routers has been disabled, as many of the attacks we will use would normally be blocked by the routers.

4.2 Testing Tools

We have two laptops to use for testing; both are running Windows 2000 workstation. These have both had VMWare installed with a copy of Red Hat Linux 7.3. The tools to be used are:

4.2.1 Ping

Ping is a very basic, but invaluable tool and comes with just about every operating system.

4.2.2 Nslookup

We will use nslookup to perform DNS queries and zone transfer attempts. Again nslookup comes bundled with most operating systems.

4.2.3 Telnet

Telnet provides terminal emulation services and will be used for part of our testing.

4.2.4 Nmap

Nmap is now a very widely known and used tool (primarily by the bad guy's unfortunately) that provides a wide range of scanning capabilities and features. We will use Nmap to scan against various ports with different TCP flags set and for OS detection through the OS fingerprinting capabilities of Nmap. (http://www.insecure.org/nmap)

4.2.5 Nessus

Nessus provides similar features to Nmap (in fact Nmap output can be fed into nessus) , but goes a step further by identifying the type of host and can optionally try and exploit vulnerabilities on open ports, whereas Nmap simply reports which ports are open. This is of course very dangerous in the wrong hands, but is an extremely valuable auditing tool.

4.2.6 *Tcpdump* (Linux) and Windump (windows)

Tcpdump is a network sniffer that we can use to capture packets for the network. This assists in verifying network activity and is a valuable troubleshooting tool. Windump is the windows version.

4.2.7 *Ethereal* **(**windows) http://www.ethereal.com

Ethereal is a GUI wrapper around the windows equivalent to Tcpdump called Windump.
We will monitor the syslog events from the PIX firewall I conjunction with our testing.

4.3 Prior to Testing

Before we commence any testing we ensure that a full backup is performed of all network devices and all servers.
Even though the network is not yet in production, we still need to obtain written permission from GIAC management before we can commence any auditing activities.

4.4 Work Effort

We have estimated that to test the entire infrastructure fully would probably take at least a week! However, as we are only testing the firewall functionality we estimate that this can be completed within 16 hours.
This will be conducted by our own internal staff with existing equipment and primarily freeware tools, so the cost is minimal. As this is not yet in production, there is no business downtime to consider so we can conduct the audit during normal business hours.

4.5 The Testing

Out test procedure is:

4.5.1 Administration Access

First we ensure that we can only connect to the firewall via SSH and HTTPs from the network management workstation. We do this by attaching a laptop to a switch off of each PIX interface, setting an IP address for the subnet off of that interface and attempting to connect to the PIX with HTTPS and SSH sessions.

4.5.2 Verify the Rule Base

Next we want to ensure that traffic we want to permit through the firewall can actually do so.

We will go through the rule set for each interface to verify that all services are reachable on the defined ports as expected. Refer to the translation and access-list section of the PIX tutorial.

Once we have verified that access is permitted to the required services we want to make sure that nothing else is permitted.

We will use one laptop attached to a switch port of each PIX interface and the second laptop to capture all traffic from the firewall. To accomplish this we have used the Cisco port spanning feature of the Ethernet switches and have connected a separate port from each switch to an Ethernet hub. We have attached our monitoring laptop to this hub. We have then spanned each of the switch ports connected to the PIX. This enables us to see all traffic from all ports of the PIX simultaneously during our testing.

4.5.3 NMAP tests

We will use Nmap as follows for TCP scans:

- nmap -sS -P0 –p <port> -vv <dest IP addres> -n –oN “output.txt”

For UDP scans we will use: nmap -sU -P0 –p <port> -vv <dest IP addres> -oN “output file.txt”

The above syntax will instruct nmap to perform a SYN stealth scan or half open connection (-sS) against the host. For this NMAP sends a packet with the SYN flag set. If the host responds with a SYN ACK packet NMAP will immediately send back a RST/ACK packet to close the connection. We will not ping the host prior to scanning (-P0 option). NMAP will by default verify the host is up before scanning, but we already know these hosts are up, so this will save us time and for the same reason we will not perform name lookups (-n option).The ‘-vv’ option means very verbose and provides more detail to us and finally we will save the nmap output to a file (-oN option).
Example:

Nmap –sS –P0 -p80 –vv 223.223.223.100 –n –oN “80.txt”

This will scan the web server at IP Address 223.223.223.100 on port 80 and will save the verbose output to file 80.txt.

4.6 From the ‘outside’ interface

1) We attach a laptop to one of the switches on the outside interface, assign it an IP address in the 223.223.223.0 /24 range and set the PIX at the default gateway. Then we run nmap to verify that the PIX is accepting connections on the permitted IP addresses and ports as expected

2) First we test that we can access the website on port 80.

Nmap output

```
# nmap (V. 3.00) scan initiated Sat Jun 14 13:23:38 2003 as:
nmap -sS -P0 -p80 -vv -n -oN c:\80.txt 223.223.223.100

Interesting ports on  (223.223.223.100):

Port       State      Service

80/tcp     open       http

# Nmap run completed at Sat Jun 14 13:24:03 2003 -- 1 IP address
(1 host up) scanned in 25 seconds
```

PIX Syslog

```
%PIX-6-302013: Built inbound TCP connection 53 for
outside:223.223.223.223/63634 (223.223.223.223/63634) to
web:192.168.5.100/80 (223.223.223.100/80)
```

3) Test that we can access the website on port 443.

Nmap output

```
# nmap (V. 3.00) scan initiated Sat Jun 14 13:58:29 2003 as:
nmap -sS -P0 -p443 -vv -n -oN c:\443.txt 223.223.223.100

Interesting ports on  (223.223.223.100):

Port       State      Service

443/tcp    open       https

# Nmap run completed at Sat Jun 14 13:59:33 2003 -- 1 IP address
(1 host up) scanned in 64 seconds
```

PIX syslog

```
%PIX-6-302013: Built inbound TCP connection 54 for
outside:223.223.223.223/59302 (223.223.223.223/59302) to
web:192.168.5.100/443 (223.223.223.100/443)
```

4) Test access to the external mail server

Nmap output

```
# nmap (V. 3.00) scan initiated Sat Jun 14 14:12:59 2003 as:
nmap -sS -P0 -p25 -vv -n -oN c:\tcp25.txt 223.223.223.12

Interesting ports on  (223.223.223.12):

Port       State      Service

25/tcp     open       smtp

# Nmap run completed at Sat Jun 14 14:13:28 2003 -- 1 IP address
(1 host up) scanned in 29 seconds
```

PIX syslog

```
%PIX-6-302013: Built inbound TCP connection 124 for
outside:223.223.223.223/54416 (223.223.223.223/54416) to
web:192.168.5.12/25 (223.223.223.12/25)
```

5) Test TCP access to the external DNS server. Note here we have used NMAP to spoof the source address to appear as the ISP DNS server source address, as these are the only permitted TCP 53 connections permitted.

Nmap output

```
# nmap (V. 3.00) scan initiated Sat Jun 14 14:16:30 2003 as:
nmap -sS -P0 -S 202.139.83.3 -p53 -vv –n -oN c:\tcp53isp.txt -e
eth0 223.223.223.50

Interesting ports on (223.223.223.50):

Port       State       Service

53/tcp     open        domain

# Nmap run completed at Sat Jun 14 14:16:59 2003 -- 1 IP address
(1 host up) scanned in 29 seconds
```

PIX Syslog

```
%PIX-6-302013: Built inbound TCP connection 132 for
outside:202.139.83.3/53726 (202.139.83.3/53726) to
web:192.168.5.50/53 (223.223.223.50/53)

Here is the PIX syslog for the second permitted ISP DNS server.

PIX-6-302013: Built inbound TCP connection 132 for
outside:202.139.83.3/53726 (202.139.83.3/53726) to
web:192.168.5.50/53 (223.223.223.50/53)
```

6) Test UDP access to the external DNS server. Note this can be from any source address. Here we use a UDP nmap scan.

Nmap output

```
# nmap (V. 3.00) scan initiated Sat Jun 14 14:10:59 2003 as:
nmap -sU -P0 -p53 -vv -n -oN c:\udp53.txt 223.223.223.50

Interesting ports on  (223.223.223.50):

Port       State       Service

53/udp     open        domain

# Nmap run completed at Sat Jun 14 14:11:26 2003 -- 1 IP address
(1 host up) scanned in 27 seconds
```

PIX Syslog

```
%PIX-6-302015: Built inbound UDP connection 150 for
outside:223.223.223.5/1457 (223.223.223.5/1457) to
web:192.168.5.50/53 (223.223.223.50/53)
```

7) Test NTP server access from the outside interface network devices. Here we set our source address to be each of the network devices. The output below is with the source IP Address set as one of the Ethernet switches.

Nmap output – NTP Server 1

```
# nmap (V. 3.00) scan initiated Sat Jun 14 14:02:18 2003 as:
nmap -sS -P0 -p123 -vv -n -oN c:\123-23.txt 223.223.223.23

Interesting ports on  (223.223.223.23):

Port       State       Service

123/tcp    open        ntp

# Nmap run completed at Sat Jun 14 14:02:47 2003 -- 1 IP address
(1 host up) scanned in 29 seconds
```

PIX Syslog – NTP Server 1

```
%PIX-6-302013: Built inbound TCP connection 109 for
outside:223.223.223.5/34304 (223.223.223.5/34304) to
inside:172.25.1.101/123 (223.223.223.23/123)
```

Nmap output – NTP Server 2

```
# nmap (V. 3.00) scan initiated Sat Jun 14 13:49:58 2003 as:
nmap -sS -P0 -p123 -vv -n -oN c:\123-24.txt 223.223.223.24

Interesting ports on  (223.223.223.24):

Port       State       Service

123/tcp    open        ntp

# Nmap run completed at Sat Jun 14 13:51:00 2003 -- 1 IP address
(1 host up) scanned in 62 seconds
```

PIX Syslog – NTP Server 2

```
%PIX-6-302013: Built inbound TCP connection 85 for
outside:223.223.223.5/46142 (223.223.223.5/46142) to
inside:172.25.2.102/123 (223.223.223.24/123)
```

8) Test Syslog server access from the outside interface network devices. Here we set our source address to be each of the network devices. The output below is with the source IP Address set as one of the Ethernet switches. We are using nmap UDP scans to test this.

Nmap output – Syslog Server 1

```
# nmap (V. 3.00) scan initiated Sat Jun 14 14:05:23 2003 as:
nmap -sU -P0 -p514 -vv -n -oN c:\udp514.txt 223.223.223.21

Interesting ports on  (223.223.223.21):

Port       State       Service

514/udp    open        syslog

# Nmap run completed at Sat Jun 14 14:06:00 2003 -- 1 IP address
(1 host up) scanned in 37 seconds
```

PIX Syslog – Syslog Server 1

```
%PIX-6-302015: Built inbound UDP connection 117 for
outside:223.223.223.5/47956 (223.223.223.5/47956) to
inside:192.168.11.11/514 (223.223.223.21/514)
```

9) Test access to TFTP server from network devices. Here we set our source address to be each of the network devices. The output below is with the source IP Address set as one of the Ethernet switches. We are using nmap UDP scans to test this.

Nmap output

```
# nmap (V. 3.00) scan initiated Mon Jun 23 07:02:31 2003 as:
nmap -sU -P0 -p69 -vv -v -T 5 -oN 69.txt 223.223.223.13

Interesting ports on  (223.223.223.13):

Port       State       Service

69/udp     open        tftp

# Nmap run completed at Mon Jun 23 07:02:38 2003 -- 1 IP address
(1 host up) scanned in 7 seconds
```

PIX Syslog

```
302015: Built inbound UDP connection 214 for
outside:223.223.223.6/40136 (223.223.223.6/40136) to
inside:192.168.11.13/69 (223.223.223.13/69)

302015: Built inbound UDP connection 215 for
outside:223.223.223.6/40137 (223.223.223.6/40137) to
inside:192.168.11.13/69 (223.223.223.13/69
```

4.7 From the vpn interface

We will perform similar tests against the vpn interface. We will configure the laptop with an IP address in each of the vpn pool ranges in turn. We will use NMAP to perform a SYN stealth scan against ports that should be open to verify that we can connect to these.

1) First we test that we can access the website on port 80 from the range of addresses assigned to the GIAC vpn pools.

Nmap output

```
# nmap (V. 3.00) scan initiated Mon Jun 23 07:51:21 2003 as:
nmap -sS -P0 -p80 -vv -n -oN 80.txt 223.223.223.100

Interesting ports on  (223.223.223.100):

Port       State       Service

80/tcp     open        http

# Nmap run completed at Mon Jun 23 07:51:22 2003 -- 1 IP address
(1 host up) scanned in 1 second
```

PIX Syslog

```
305009: Built static translation from web:192.168.5.100 to
vpn:223.223.223.100
```

```
302013: Built inbound TCP connection 18 for
vpn:10.10.2.100/42286 (10.10.2.100/42286) to
web:192.168.5.100/80 (223.223.223.100/80)
```

2) Then we test that we can access the website on port 443

Nmap output

```
# nmap (V. 3.00) scan initiated Mon Jun 23 07:51:57 2003 as:
nmap -sS -P0 -p443 -vv -n -oN 443.txt 223.223.223.100

Interesting ports on  (223.223.223.100):

Port       State       Service

443/tcp     open       https

# Nmap run completed at Mon Jun 23 07:51:57 2003 -- 1 IP address
(1 host up) scanned in 0 seconds
```

PIX Syslog

```
02013: Built inbound TCP connection 19 for vpn:10.10.2.100/43037
(10.10.2.100/43037) to web:192.168.5.100/443
(223.223.223.100/443
```

3) Then we test that we can access the internal DNS server

Nmap output

```
# nmap (V. 3.00) scan initiated Mon Jun 23 07:52:57 2003 as:
nmap -sU -P0 -p53 -vv -n -oN 53.txt 172.25.1.50

Interesting ports on  (172.25.1.50):

Port       State       Service

53/udp     open        domain

# Nmap run completed at Mon Jun 23 07:52:58 2003 -- 1 IP address
(1 host up) scanned in 1 second
```

PIX Syslog

```
305009: Built static translation from inside:172.25.1.50 to
vpn:172.25.1.50

302015: Built inbound UDP connection 20 for
vpn:10.10.2.100/63142 (10.10.2.100/63142) to
inside:172.25.1.50/53 (172.25.1.50/53)
```

4) Then we test that the vpn network devices can send messages to the syslog servers

Nmap output

```
# nmap (V. 3.00) scan initiated Mon Jun 23 07:45:57 2003 as:
nmap -sU -P0 -p514 -vv -n -oN 514.txt 192.168.11.11

Interesting ports on  (192.168.11.11):

Port       State       Service
```

```
514/udp    open        syslog

# Nmap run completed at Mon Jun 23 07:45:58 2003 -- 1 IP address
(1 host up) scanned in 1 second
```

PIX Syslog

```
305009: Built static translation from inside:192.168.11.11 to
vpn:192.168.11.11

302015: Built inbound UDP connection 12 for
vpn:192.168.3.3/52235 (192.168.3.3/52235) to
inside:192.168.11.11/514 (192.168.11.11/514)
```

5) Test access to the internal mail server from the range of addresses assigned to the GIAC vpn pools.

Nmap output

```
# nmap (V. 3.00) scan initiated Mon Jun 23 07:53:45 2003 as:
nmap -sS -P0 -p25 -vv -n -oN 25.txt 172.25.2.50

Interesting ports on  (172.25.2.50):

Port       State       Service

25/tcp     open         smtp

# Nmap run completed at Mon Jun 23 07:53:48 2003 -- 1 IP address
(1 host up) scanned in 3 seconds
```

PIX Syslog

```
609001: Built local-host inside:172.25.2.50

305009: Built static translation from inside:172.25.2.50 to
vpn:172.25.2.50

302013: Built inbound TCP connection 22 for
vpn:10.10.2.100/51316 (10.10.2.100/51316) to
inside:172.25.2.50/25 (172.25.2.50/25)
```

This was also tested with a source address from the other GIAC VPN ranges, which all connected successfully.

6) Test access to the partsup ftp server. For this we test with a source address from the partsup vpn pool range, the GIAC admin range and the GIAC cookie processing range.

Nmap output

```
# nmap (V. 3.00) scan initiated Mon Jun 23 07:55:25 2003 as:
nmap -sS -P0 -p21 -vv -n -oN 21.txt 192.168.4.11

Interesting ports on  (192.168.4.11):

Port       State       Service

21/tcp     open         ftp

# Nmap run completed at Mon Jun 23 07:55:28 2003 -- 1 IP address
(1 host up) scanned in 3 seconds
```

PIX Syslog

```
609001: Built local-host partsup:192.168.4.11

305009: Built static translation from partsup:192.168.4.11 to
vpn:192.168.4.11

302013: Built inbound TCP connection 28 for
vpn:10.10.2.100/34562 (10.10.2.100/34562) to
partsup:192.168.4.11/21 (192.168.4.11/21)
```

This was also tested successfully from the GIAC admin vpn pool range and the partner and supplier vpn pool ranges.

7) Test access to the partsup SQL server. For this we test with a source address from the partsup vpn pool range, the GIAC admin range and the GIAC cookie processing range.

Nmap output

```
# nmap (V. 3.00) scan initiated Mon Jun 23 07:56:14 2003 as:
nmap -sS -P0 -p1521 -vv -n -oN 1521.txt 192.168.4.10

Interesting ports on  (192.168.4.10):

Port       State      Service

1521/tcp   open       oracle

# Nmap run completed at Mon Jun 23 07:56:17 2003 -- 1 IP address
(1 host up) scanned in 3 seconds
```

PIX Syslog

```
609001: Built local-host partsup:192.168.4.10

305009: Built static translation from partsup:192.168.4.10 to
vpn:192.168.4.10

302013: Built inbound TCP connection 34 for
vpn:10.10.2.100/37779 (10.10.2.100/37779) to
partsup:192.168.4.10/1521 (192.168.4.10/1521)
```

This was also tested successfully from the GIAC admin vpn pool range and the partner and supplier vpn pool ranges.

8) Test that the vpn network devices can connect to the ntp servers

Nmap output

```
# nmap (V. 3.00) scan initiated Mon Jun 23 07:47:09 2003 as:
nmap -sS -P0 -p123 -vv -n -oN 123.txt 172.25.1.101

Interesting ports on  (172.25.1.101):

Port       State      Service

123/tcp    open       ntp

# Nmap run completed at Mon Jun 23 07:47:09 2003 -- 1 IP address
(1 host up) scanned in 0 seconds
```

PIX Syslog

```
302013: Built inbound TCP connection 15 for
vpn:192.168.3.3/44660 (192.168.3.3/44660) to
inside:172.25.1.101/123 (172.25.1.101/123)
```

9) Test that the vpn network devices can send configurations to the TFTP server.

Nmap output

```
# nmap (V. 3.00) scan initiated Mon Jun 23 07:48:00 2003 as:
nmap -sU -P0 -p69 -vv -n -T 5 -oN 69.txt 192.168.11.13

Interesting ports on  (192.168.11.13):

Port       State       Service

69/udp     open        tftp

# Nmap run completed at Mon Jun 23 07:48:02 2003 -- 1 IP address
(1 host up) scanned in 2 seconds
```

PIX Syslog

```
305009: Built static translation from inside:192.168.11.13 to
vpn:192.168.11.13

302015: Built inbound UDP connection 16 for
vpn:192.168.3.3/56973 (192.168.3.3/56973) to
inside:192.168.11.13/69 (192.168.11.13/69)
```

4.8 From the web interface

We now configure the laptop with an IP address in the web interface range. We will use NMAP to perform a SYN stealth scan against ports that should be open to verify that we can connect to these.

1) Test access from the web servers to the Websphere application servers. For this we test with a source address of 192.168.5.200, which is the IP used by the web servers source group of the content switches. All outbound web server connections will use this source address. First we test to port 9080

Nmap output

```
# nmap (V. 3.00) scan initiated Thu Jun 19 08:05:43 2003 as:
nmap -sT -P0 -p9080 -vv -n -oN 9080.txt 192.168.9.10

Interesting ports on  (192.168.9.10):

Port       State       Service

9080/tcp   open        unknown

# Nmap run completed at Thu Jun 19 08:05:45 2003 -- 1 IP address
(1 host up) scanned in 2 seconds
```

PIX Syslog

```
302013: Built inbound TCP connection 361 for
web:192.168.5.200/3158 (192.168.5.200/3158) to
inside:192.168.9.10/9080 (192.168.9.10/9080)
```

2) Then we test to port 9443 to the Websphere servers.

Nmap output

```
# nmap (V. 3.00) scan initiated Thu Jun 19 08:06:38 2003 as:
nmap -sT -P0 -p9443 -vv -n -oN 9443.txt 192.168.9.10

Interesting ports on  (192.168.9.10):

Port       State       Service

9443/tcp   open         unknown

# Nmap run completed at Thu Jun 19 08:06:39 2003 -- 1 IP address
(1 host up) scanned in 1 second
```

PIX Syslog

```
302013: Built inbound TCP connection 371 for
web:192.168.5.200/3166 (192.168.5.200/3166) to
inside:192.168.9.10/9443 (192.168.9.10/9443)
```

The tests are repeated with identical results for the second Websphere server.

3) Test that the web servers can talk to the LDAP server.

Nmap output

```
# nmap (V. 3.00) scan initiated Mon Jun 23 07:05:35 2003 as:
nmap -sS -P0 -p389 -vv -v -oN 389.txt 192.168.9.100

Interesting ports on  (192.168.9.100):

Port       State       Service

389/tcp    open           ldap

# Nmap run completed at Mon Jun 23 07:05:43 2003 -- 1 IP address
(1 host up) scanned in 8 seconds
```

PIX Syslog

```
305009: Built static translation from inside:192.168.9.100 to
web:192.168.9.100

302013: Built inbound TCP connection 216 for
web:192.168.5.200/37078 (192.168.5.200/37078) to
inside:192.168.9.100/389 (192.168.9.100/389)
```

4) Test that the web segment network devices can send syslog messages through the firewall.

Nmap output

```
# nmap (V. 3.00) scan initiated Thu Jun 19 07:32:26 2003 as:
nmap -sU -P0 -p514 -vv -n -oN 514.txt 192.168.11.11
```

```
Interesting ports on  (192.168.11.11):

Port       State       Service

514/udp    open        syslog

# Nmap run completed at Thu Jun 19 07:32:27 2003 -- 1 IP address
(1 host up) scanned in 1 second
```

PIX Syslog

```
302013:302015: Built inbound UDP connection 257 for
web:192.168.5.4/35340 (192.168.5.4/35340) to
inside:192.168.11.11/514 (192.168.11.11/514)
```

The same results were seen for the second syslog server.

5) Test that the external DNS server can send DNS lookups

Nmap output

```
# nmap (V. 3.00) scan initiated Thu Jun 19 07:45:05 2003 as:
nmap -sU -P0 -p53 -vv -n -oN 53.txt 202.139.83.3

Interesting ports on  (202.139.83.3):

Port       State       Service

53/udp     open        domain

# Nmap run completed at Thu Jun 19 07:45:06 2003 -- 1 IP address
(1 host up) scanned in 1 second
```

PIX Syslog

```
302015:    Built    outbound    UDP    connection    266    for
outside:202.139.83.3/53          (202.139.83.3/53)          to
web:192.168.5.50/62758 (223.223.223.50/62758)
```

6) Test that the external DNS server can perform zone transfers to the ISP DNS servers.

Nmap output

```
# nmap (V. 3.00) scan initiated Thu Jun 19 07:46:46 2003 as:
nmap -sT -P0 -p53 -vv -n -oN ispdns.txt 202.139.83.3

Interesting ports on  (202.139.83.3):

Port       State       Service

53/tcp     open        domain

# Nmap run completed at Thu Jun 19 07:46:48 2003 -- 1 IP address
(1 host up) scanned in 2 seconds
```

PIX Syslog

```
302013: Built outbound TCP connection 278 for
outside:202.139.83.3/53 (202.139.83.3/53) to
web:192.168.5.50/3028 (223.223.223.50/3028)
```

7) Test that the external mail server can send outbound mail.

Nmap output

```
# nmap (V. 3.00) scan initiated Thu Jun 19 07:52:38 2003 as:
nmap -sT -P0 -p25 -vv -n –oN 25.txt 192.65.90.202

Interesting ports on  (192.65.90.202):

Port       State       Service

25/tcp     open        smtp

# Nmap run completed at Thu Jun 19 07:52:40 2003 -- 1 IP address
(1 host up) scanned in 2 seconds
```

PIX Syslog

```
302013: Built outbound TCP connection 301 for
outside:192.65.90.202/25 (192.65.90.202/25) to
web:192.168.5.12/3083 (223.223.223.12/3083)
```

8) Test that the web segment network devices can communicate with the NTP servers.

Nmap output

```
# nmap (V. 3.00) scan initiated Thu Jun 19 07:31:37 2003 as:
nmap -sT -P0 -p123 -vv -n -oN 123-2.txt -T 5 172.25.2.102

Interesting ports on  (172.25.2.102):

Port       State       Service

123/tcp    open        ntp

# Nmap run completed at Thu Jun 19 07:31:39 2003 -- 1 IP address
(1 host up) scanned in 2 seconds
```

PIX Syslog

```
302013: Built inbound TCP connection 256 for
web:192.168.5.4/3001 (192.168.5.4/3001) to
inside:172.25.2.102/123 (172.25.2.102/123)
```

9) Test that the web segment network devices can communicate with the TFTP configuration server.

Nmap output

```
# nmap (V. 3.00) scan initiated Fri Jun 20 07:43:16 2003 as:
nmap -sU -P0 -p69 -vv -n -oN 69.txt 192.168.11.13

Interesting ports on  (192.168.11.13):

Port       State       Service

69/udp     open        tftp

# Nmap run completed at Fri Jun 20 07:43:18 2003 -- 1 IP address
(1 host up) scanned in 2 seconds
```

PIX Syslog

```
302015: Built inbound UDP connection 210 for
web:192.168.5.3/57385 (192.168.5.3/57385) to
inside:192.168.11.13/69 (192.168.11.13/69)
```

4.9 From the partsup interface

We now configure our laptop in the partsup IP address range and run similar tests form this interface of the firewall.

There should not be any traffic initiated from this segment, other than syslog, NTP and TFTP for the network devices.

1) Test that the partsup segment network devices can communicate with the NTP servers.

Nmap output

```
# nmap (V. 3.00) scan initiated Thu Jun 19 07:20:07 2003 as:
nmap -sT -P0 -p123 -vv -n -oN 123-2.txt -T 5 172.25.2.102

Interesting ports on  (172.25.2.102):

Port       State       Service

123/tcp    open        ntp

# Nmap run completed at Thu Jun 19 07:20:09 2003 -- 1 IP address
(1 host up) scanned in 2 seconds
```

PIX Syslog

```
302013: Built inbound TCP connection 223 for
partsup:192.168.4.3/2950 (192.168.4.3/2950) to
inside:172.25.2.102/123 (172.25.2.102/123)
```

2) Test that the partsup segment network devices can communicate with the syslog servers

Nmap output

```
# nmap (V. 3.00) scan initiated Thu Jun 19 07:19:32 2003 as:
nmap -sU -P0 -p514 -vv -n -oN 514.txt 192.168.11.11

Interesting ports on  (192.168.11.11):

Port       State       Service

514/udp    open        syslog

# Nmap run completed at Thu Jun 19 07:19:33 2003 -- 1 IP address
(1 host up) scanned in 1 second
```

PIX Syslog

```
302015: Built inbound UDP connection 213 for
partsup:192.168.4.3/57367 (192.168.4.3/57367) to
inside:192.168.11.11/514 (192.168.11.11/514)
```

3) Test that the partsup segment network devices can communicate with the TFTP configuration server.

Nmap output

```
# nmap (V. 3.00) scan initiated Mon Jun 23 07:09:50 2003 as:
nmap -sU -P0 -p69 -vv -v -oN 69.txt 192.168.11.13

Interesting ports on  (192.168.11.13):

Port       State       Service

69/udp     open        tftp

# Nmap run completed at Mon Jun 23 07:09:57 2003 -- 1 IP address
(1 host up) scanned in 7 seconds
```

PIX Syslog

```
305009: Built static translation from inside:192.168.11.13 to
partsup:192.168.11.13

302015: Built inbound UDP connection 222 for
partsup:192.168.4.3/49283 (192.168.4.3/49283) to
inside:192.168.11.13/69 (192.168.11.13/69)
```

4.10 From the inside interface

Next we move the laptop to the inside interface of our firewall and run a range of tests against this interface.

1) Test outbound http access from the proxy server. Here we have attached the laptop to the proxy server screened subnet off of the Checkpoint firewall and set our source address to be that of the proxy server.

Nmap output

```
# nmap (V. 3.00) scan initiated Fri Jun 20 07:07:23 2003 as:
nmap -sS -P0 -p80 -vv -n –oN 80.txt 209.76.89.5

Interesting ports on  (209.76.89.5):

Port       State       Service

80/tcp     open        http

# Nmap run completed at Fri Jun 20 07:07:27 2003 -- 1 IP address
(1 host up) scanned in 4 seconds
```

PIX Syslog

```
305011: Built dynamic TCP translation from
inside:192.168.9.50/46690 to outside:223.223.223.7/1065

302013: Built outbound TCP connection 173 for
outside:209.76.89.5/80 (209.76.89.5/80) to
inside:192.168.9.50/46690 (223.223.223.7/1065)
```

2) Test outbound SSL access from the proxy server. Again our source address is configured to be that of the proxy server.

Nmap output

```
# nmap (V. 3.00) scan initiated Fri Jun 20 07:07:42 2003 as:
nmap -sS -P0 -p443 -vv -n –oN 443.txt 209.76.89.5

Interesting ports on  (209.76.89.5):

Port       State       Service

443/tcp    open        https

# Nmap run completed at Fri Jun 20 07:07:46 2003 -- 1 IP address
(1 host up) scanned in 4 seconds
```

PIX Syslog

```
305011: Built dynamic TCP translation from
inside:192.168.9.50/38088 to outside:223.223.223.7/1077

302013: Built outbound TCP connection 185 for
outside:209.76.89.5/443 (209.76.89.5/443) to
inside:192.168.9.50/38088 (223.223.223.7/1077)
```

3) Test outbound NS lookups from internal DNS server.

Nmap output

```
# nmap (V. 3.00) scan initiated Fri Jun 20 07:20:19 2003 as:
nmap -sU -P0 -p53 -vv -n –oN 53.txt 192.168.5.50

Interesting ports on  (192.168.5.50):

Port       State       Service

53/udp     open        domain

# Nmap run completed at Fri Jun 20 07:20:20 2003 -- 1 IP address
(1 host up) scanned in 1 second
```

PIX Syslog

```
302015: Built outbound UDP connection 199 for
web:192.168.5.50/53 (192.168.5.50/53) to
inside:172.25.1.50/42680 (172.25.1.50/42680)
```

4) Test outbound FTP from the proxy server.

Nmap output

```
# nmap (V. 3.00) scan initiated Fri Jun 20 07:08:37 2003 as:
nmap -sS -P0 -p21 -vv -n –oN 21.txt 209.76.89.5

Interesting ports on  (209.76.89.5):

Port       State       Service

21/tcp     open        ftp

# Nmap run completed at Fri Jun 20 07:08:40 2003 -- 1 IP address
(1 host up) scanned in 3 seconds
```

PIX Syslog

```
305011: Built dynamic TCP translation from
inside:192.168.9.50/46471 to outside:223.223.223.7/1083

302013: Built outbound TCP connection 191 for
outside:209.76.89.5/21 (209.76.89.5/21) to
inside:192.168.9.50/46471 (223.223.223.7/1083)
```

5) Test outbound LDAP requests from the internal hosts

Nmap output

```
# nmap (V. 3.00) scan initiated Thu Jun 19 08:28:17 2003 as:
nmap -sT -P0 -p389 -vv -n –oN 389.txt 202.139.8.83

Interesting ports on  (202.139.8.83):

Port       State       Service

389/tcp    open           ldap

# Nmap run completed at Thu Jun 19 08:28:19 2003 -- 1 IP address
(1 host up) scanned in 2 seconds
```

PIX Syslog

```
305011: Built dynamic TCP translation from
inside:172.25.100.200/3277 to outside:223.223.223.7/1047

302013: Built outbound TCP connection 23 for
outside:202.139.8.83/389 (202.139.8.83/389) to
inside:172.25.100.200/3277 (223.223.223.7/1047)
```

During testing we came across a problem with PIX crashing and reloading from any LDAP requests. It appears the ASA algorithm of the PIX identifies the LDAP as an ILS (Microsoft Internet Locator Service) request and attempts to NAT part of the payload. The workaround is to disable the fixup for ILS or to upgrade to PIX software version 6.3. This is detailed in Cisco bug id's CSCdx78331 and CSCdx73007.

6) Test outbound POP3 requests from the internal hosts

Nmap output

```
# nmap (V. 3.00) scan initiated Thu Jun 19 08:29:56 2003 as:
nmap -sT -P0 -p110 -vv -n –oN 110.txt 202.139.8.83

Interesting ports on  (202.139.8.83):

Port       State       Service

110/tcp    open        pop-3

# Nmap run completed at Thu Jun 19 08:29:57 2003 -- 1 IP address
(1 host up) scanned in 1 second
```

PIX Syslog

```
305011: Built dynamic TCP translation from
inside:172.25.100.200/3285 to outside:223.223.223.7/1053
```

```
302013: Built outbound TCP connection 29 for
outside:202.139.8.83/110 (202.139.8.83/110) to
inside:172.25.100.200/3285 (223.223.223.7/1053)
```

7) Test outbound news requests from the internal hosts

Nmap output

```
# nmap (V. 3.00) scan initiated Mon Jun 23 07:23:50 2003 as:
nmap -sS -P0 -p119 -vv -v -oN 119.txt 203.202.98.9

Interesting ports on  (203.202.98.9):

Port       State       Service

119/tcp    open           nntp

# Nmap run completed at Mon Jun 23 07:23:59 2003 -- 1 IP address
(1 host up) scanned in 9 seconds
```

PIX Syslog

```
305011: Built dynamic TCP translation from
inside:172.25.100.250/59011 to outside:223.223.223.7/1027

302013: Built outbound TCP connection 3 for
outside:203.202.98.9/119 (203.202.98.9/119) to
inside:172.25.100.250/59011 (223.223.223.7/1027)
```

8) Test mail from internal mail server to external mail server

Nmap output

```
# nmap (V. 3.00) scan initiated Fri Jun 20 07:13:05 2003 as:
nmap -sS -P0 -p25 -vv -n -oN 25.txt 192.168.5.12

Interesting ports on  (192.168.5.12):

Port       State       Service

25/tcp     open        smtp

# Nmap run completed at Fri Jun 20 07:13:08 2003 -- 1 IP address
(1 host up) scanned in 3 seconds
```

PIX Syslog

```
609001: Built local-host inside:172.25.2.50

305009: Built static translation from inside:172.25.2.50 to
web:172.25.2.50
```

9) Test the connection from internal hosts to replica SQL server on partsup screened subnet.

Nmap output

```
# nmap (V. 3.00) scan initiated Thu Jun 19 08:40:28 2003 as:
nmap -sT -P0 -p1521 -vv -n -oN 1521.txt 192.168.4.10

Interesting ports on  (192.168.4.10):
```

```
Port       State       Service

1521/tcp   open        oracle

# Nmap run completed at Thu Jun 19 08:40:30 2003 -- 1 IP address
(1 host up) scanned in 2 seconds
```

PIX Syslog

```
302013: Built outbound TCP connection 41 for
partsup:192.168.4.10/1521 (192.168.4.10/1521) to
inside:172.25.100.200/3342 (172.25.100.200/3342)
```

10) Test the connections from internal hosts to the web management port 8081 of the web servers.

Nmap output

```
# nmap (V. 3.00) scan initiated Thu Jun 19 08:44:03 2003 as:
nmap -sT -P0 -p8081 -vv -n -oN 8081.txt 192.168.5.101

Interesting ports on  (192.168.5.101):

Port       State       Service

8081/tcp   open        blackice-icecap

# Nmap run completed at Thu Jun 19 08:44:05 2003 -- 1 IP address
(1 host up) scanned in 2 seconds
```

PIX Syslog

```
302013: Built outbound TCP connection 47 for
web:192.168.5.101/8081 (192.168.5.101/8081) to
inside:172.25.100.200/3358 (172.25.100.200/3358)
```

11) Test connections from internal hosts to the web management port 9173 of the web servers

Nmap output

```
# nmap (V. 3.00) scan initiated Thu Jun 19 08:46:34 2003 as:
nmap -sT -P0 -p9173 -vv -n -oN 9173.txt 192.168.5.101

Interesting ports on  (192.168.5.101):

Port       State       Service

9173/tcp   open        unknown

# Nmap run completed at Thu Jun 19 08:46:36 2003 -- 1 IP address
(1 host up) scanned in 2 seconds
```

PIX Syslog

```
302013: Built outbound TCP connection 77 for
web:192.168.5.101/9173 (192.168.5.101/9173) to
inside:172.25.100.200/3398 (172.25.100.200/3398)
```

12) Test connections from internal hosts to the web management port 16187 of the web servers

Nmap output

```
# nmap (V. 3.00) scan initiated Thu Jun 19 08:52:40 2003 as:
nmap -sT -P0 -p16187 -vv -n –oN 16187.txt 192.168.5.101

Interesting ports on  (192.168.5.101):

Port       State       Service

16187/tcp  open        unknown

# Nmap run completed at Thu Jun 19 08:52:42 2003 -- 1 IP address
(1 host up) scanned in 2 seconds
```

PIX Syslog

```
302013: Built outbound TCP connection 144 for
web:192.168.5.101/16187 (192.168.5.101/16187) to
inside:172.25.100.200/3483 (172.25.100.200/3483)
```

13) Test Telnet from the proxy server to the network devices.

Nmap output

```
# nmap (V. 3.00) scan initiated Thu Jun 19 08:36:43 2003 as:
nmap -sT -P0 -p23 -vv -n –oN 23p.txt 192.168.5.3

Interesting ports on  (192.168.5.3):

Port       State       Service

23/tcp     open        telnet

# Nmap run completed at Thu Jun 19 08:36:45 2003 -- 1 IP address
(1 host up) scanned in 2 seconds
```

PIX Syslog

```
302013: Built outbound TCP connection 207 for web:192.168.5.3/22
(192.168.5.3/22) to inside:192.168.11.50/63183
(192.168.11.50/63183)
```

14) Test outbound telnet from the internal hosts

Nmap output

```
# nmap (V. 3.00) scan initiated Thu Jun 19 08:38:41 2003 as:
nmap -sT -P0 -p23 -vv -n –oN 23i.txt 202.202.3.67

Interesting ports on  (202.202.3.67):

Port       State       Service

23/tcp     open        telnet


# Nmap run completed at Thu Jun 19 08:38:41 2003 -- 1 IP address
(1 host up) scanned in 2 seconds
```

PIX Syslog

```
305011: Built dynamic TCP translation from
inside:172.25.100.200/3263 to outside:223.223.223.7/1035

302013: Built outbound TCP connection 11 for
outside:202.139.8.83/23 (202.139.8.83/23) to
inside:172.25.100.200/3263 (223.223.223.7/1035)
```

4.11 Port Sweeps

We have verified that all required connections are permitted through the firewall. We need to be sure that we do not have any unexpected open ports. To verify this, we perform a scan against all ports from each interface on the PIX.

For example, the following NMAP command was used to scan all ports against the IP address of the web site from the vpn interface:

Nmap –sT –P0 –p 1-65535 –n –vv –oN “open.txt” 223.223.223.0/24

No unexpected open ports were found, and the extract from the PIX syslog shows packets being dropped from the vpn interface during one of the above scans. This will show up the open ports that we have tested above, but the above command is an easy way to scan an entire subnet for open ports, although this can take several hours to complete!

```
106023: Deny tcp src vpn:10.10.2.100/4343 dst
web:223.223.223.100/48185 by access-group "vpn_access_in"

106023: Deny tcp src vpn:10.10.2.100/4344 dst
web:223.223.223.100/51481 by access-group "vpn_access_in"

106023: Deny tcp src vpn:10.10.2.100/4345 dst
web:223.223.223.100/51303 by access-group "vpn_access_in"

106023: Deny tcp src vpn:10.10.2.100/4346 dst
web:223.223.223.100/5114 by access-group "vpn_access_in"

106023: Deny tcp src vpn:10.10.2.100/4347 dst
web:223.223.223.100/12071 by access-group "vpn_access_in"

106023: Deny tcp src vpn:10.10.2.100/4348 dst
web:223.223.223.100/48185 by access-group "vpn_access_in"

106023: Deny tcp src vpn:10.10.2.100/4349 dst
web:223.223.223.100/51481 by access-group "vpn_access_in"

106023: Deny tcp src vpn:10.10.2.100/4350 dst
web:223.223.223.100/51303 by access-group "vpn_access_in"

106023: Deny tcp src vpn:10.10.2.100/4351 dst
web:223.223.223.100/5114 by access-group "vpn_access_in"

106023: Deny tcp src vpn:10.10.2.100/4352 dst
web:223.223.223.100/58939 by access-group "vpn_access_in"

106023: Deny tcp src vpn:10.10.2.100/4353 dst
web:223.223.223.100/58198 by access-group "vpn_access_in"
```

```
106023: Deny tcp src vpn:10.10.2.100/4354 dst
web:223.223.223.100/35018 by access-group "vpn_access_in"

106023: Deny tcp src vpn:10.10.2.100/4355 dst
web:223.223.223.100/9572 by access-group "vpn_access_in"

106023: Deny tcp src vpn:10.10.2.100/4356 dst
web:223.223.223.100/51848 by access-group "vpn_access_in"

106023: Deny tcp src vpn:10.10.2.100/4357 dst
web:223.223.223.100/58939 by access-group "vpn_access_in"

106023: Deny tcp src vpn:10.10.2.100/4358 dst
web:223.223.223.100/58198 by access-group "vpn_access_in"

106023: Deny tcp src vpn:10.10.2.100/4359 dst
web:223.223.223.100/35018 by access-group "vpn_access_in"
```

Similar NMAP commands for each local interface address were used against every interface of the PIX.

We did not find any unexpected open ports, which is good.

4.12 Spoofed Addresses

We will now run another test to an open port, but with a spoofed source IP address to ensure that the firewall does not accept the connection. Here we use the NMAP –S option to set out spoofed source IP address.

Nmap output

```
# nmap (V. 3.00) scan initiated Sat Jun 14 15:51:22 2003 as:
nmap -sS -P0 -p80 -vv -S 23.1.1.1 -n -e eth0 -oN spoofed.txt
223.223.223.100

Interesting ports on  (223.223.223.100):

Port       State      Service

80/tcp     filtered   http

# Nmap run completed at Sat Jun 14 15:51:50 2003 -- 1 IP address
(1 host up) scanned in 28 seconds
```

PIX Syslog

```
%PIX-4-106023: Deny tcp src outside:23.1.1.1/59707 dst
web:223.223.223.100/80 by access-group "outside_access_in"
```

The above test was performed from the outside interface, for which we have configured an access-list against a range of illegal and reserved source IP addresses.

We will now run similar tests against each interface of the PIX. The difference here is that we are now using the 'verify reverse path' feature of the PIX to block any source addresses that the PIX does not have a valid route for against the interface on which the packet was received. We do not need to use the same access-list on the outside interface as we only have specific routes via the other interfaces, which include our internal RFC1918 address ranges.

The following shows an NMAP scan with a spoofed address against the vpn interface.

Nmap output

```
# nmap (V. 3.00) scan initiated Mon Jun 23 07:57:43 2003 as:
nmap -sS -P0 -p1521 -vv -n -S 21.2.4.5 -e eth0 192.168.4.10

Interesting ports on  (192.168.4.10):

Port       State       Service

1521/tcp   filtered    oracle

# Nmap run completed at Mon Jun 23 07:57:46 2003 -- 1 IP address
(1 host up) scanned in 3 seconds
```

PIX Syslog

```
106021: Deny tcp reverse path check from 21.2.4.5 to
192.168.4.10 on interface vpn
```

We are required to use an access-list on the outside interface as our default route is out of that interface. As this will match any source address, the ‘verify reverse path’ feature will not work.

4.13 TCP Attacks

Now want to verify the firewalls stateful inspection capabilities to ensure that the firewall cannot be bypassed by setting various TCP options and flags. To do this we will scan against valid ports with a valid IP address, but we will set various TCP flags to see if the firewall will accept packets that do not have the SYN flag set and are not part of an existing connection. We will use the laptop running ethereal to monitor the web screened subnet to see if the firewall actually passes any traffic through. We will run the following scans.

Fin Scan

This set the FIN flag in the TCP packet.

```
# nmap (V. 3.00) scan initiated Sat Jun 14 15:59:28 2003 as:
nmap -sF -P0 -p80 -vv -n -e eth0 223.223.223.100 -oN fin.txt

Interesting ports on  (223.223.223.100):

Port       State       Service

80/tcp     open        http

# Nmap run completed at Sat Jun 14 15:59:34 2003 -- 1 IP address
(1 host up) scanned in 6 seconds
```

Note here, that NMAP reports the port as open, but we can see that the PIX firewall actually drops this packet.

```
%PIX-6-106015: Deny TCP (no connection) from 223.223.223.5/59523
to 223.223.223.100/80 flags FIN on interface outside
```

Null Scan

This scan sends TCP packets without ANY of the TCP options set.

```
# nmap (V. 3.00) scan initiated Sat Jun 14 16:01:30 2003 as:
nmap -sN -P0 -p80 -vv -n -e eth0 223.223.223.100 –oN null.txt

Interesting ports on  (223.223.223.100):

Port       State       Service

80/tcp     open        http

# Nmap run completed at Sat Jun 14 16:01:57 2003 -- 1 IP address
(1 host up) scanned in 27 seconds
```

Again, NMAP reports the port as open, but the PIX firewall drops the packet.

```
%PIX-6-106015: Deny TCP (no connection) from 223.223.223.5/58288
to 223.223.223.21/80 flags  on interface outside
```

Ack Scan

This sends TCP packets with the ACK flags set.

```
# nmap (V. 3.00) scan initiated Sat Jun 14 16:03:11 2003 as:
nmap -sA -P0 -p80 -vv -n -e eth0 223.223.223.100 –oN ack.txt

Interesting ports on  (223.223.223.100):

Port       State       Service

80/tcp     filtered    http

# Nmap run completed at Sat Jun 14 16:03:40 2003 -- 1 IP address
(1 host up) scanned in 29 seconds
```

This time NMAP reports the port as filtered and the PIX drops the packet.

```
%PIX-6-106015: Deny TCP (no connection) from 223.223.223.5/38832
to 223.223.223.21/80 flags ACK  on interface outside
```

XMAS Tree Scan

This sets the FIN, URG and PUSH flags in the TCP packet.

```
# nmap (V. 3.00) scan initiated Sat Jun 14 16:05:31 2003 as:
nmap -sX -P0 -p80 -vv -n -e eth0 223.223.223.100 –oN xmas.txt

Interesting ports on  (223.223.223.100):

Port       State       Service

80/tcp     open        http

# Nmap run completed at Sat Jun 14 16:05:58 2003 -- 1 IP address
(1 host up) scanned in 27 seconds
```

Again, NMAP reports the port as open, but the PIX firewall drops the packet.

```
%PIX-6-106015: Deny TCP (no connection) from 223.223.223.5/39261
to 223.223.223.100/80 flags FIN PSH URG  on interface outside
```

4.14 Fingerprint the PIX

At attempt to run an OS fingerprint scan against the PIX did not yield much.

```
C:\>nmap -sT -P0 -p80 -vv -n 223.223.223.7 -T5 -O

Starting nmap V. 3.00 ( www.insecure.org/nmap )
Host  (223.223.223.7) appears to be up ... good.
Initiating Connect() Scan against  (223.223.223.7)
The Connect() Scan took 2 seconds to scan 1 ports.
Warning:  OS detection will be MUCH less reliable because we did
not find at lea
st 1 open and 1 closed TCP port
Interesting ports on  (223.223.223.7):
Port       State       Service
80/tcp     filtered    http
Too many fingerprints match this host for me to give an accurate
OS guess
TCP/IP fingerprint:
SInfo(V=3.00%P=i686-pc-windows-windows%D=7/14%Time=3F11CA59%O=-
1%C=-1)
T5(Resp=N)
T6(Resp=N)
T7(Resp=N)
PU(Resp=N)

Nmap run completed -- 1 IP address (1 host up) scanned in 15
seconds
```

From the output above NMAP was not able to identify the firewall.

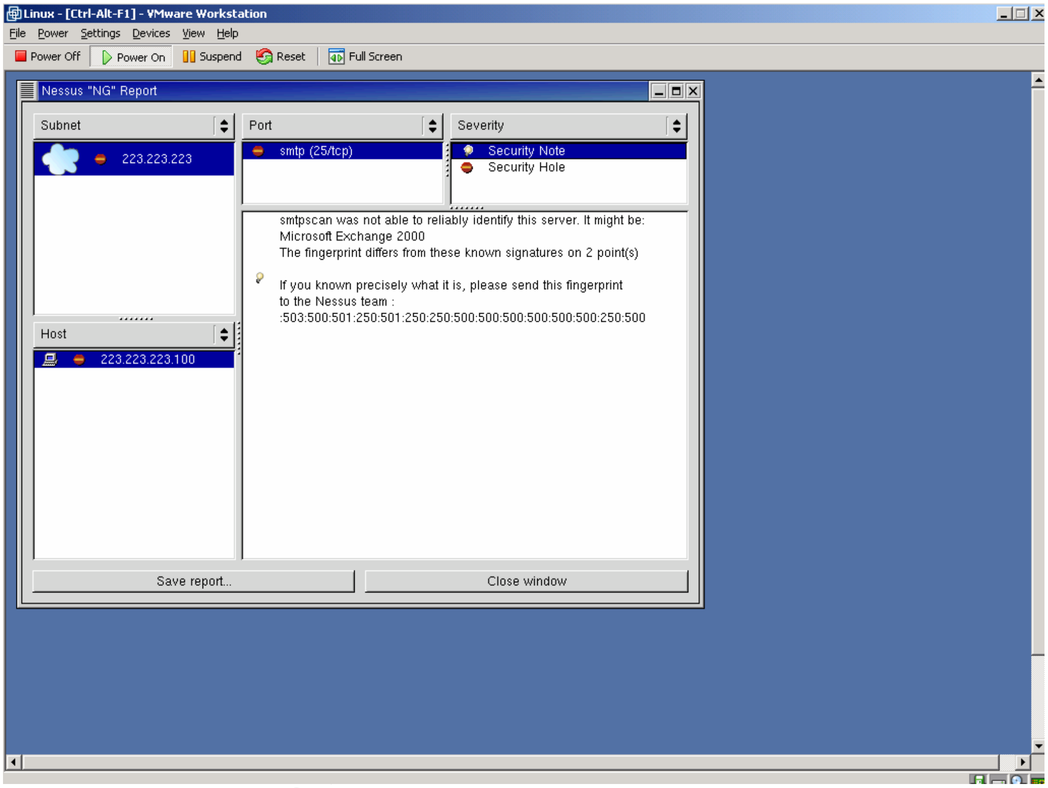
4.15 PIX SYSLOG

The PIX syslog has been a valuable tool during our auditing phase and has been used to verify all of our NMAP scans. We have also seen that even though NMAP alarmingly reported some ports as open during our illegal TCP scans, the PIX did actually reject those packets.

4.16 Nessus Attacks

The following screenshots show some examples of attacks with the Nessus tool that can be used to identify and exploit possible vulnerabilities. Nessus is an extremely powerful tool and will test against a range of known security vulnerabilities. Additionally users can add their own custom tests with the Nessus Attack Scripting Language.

The following two screenshots shows the Nessus report for a scan against a Mail server. In this example we scanned an Exchange Mail.



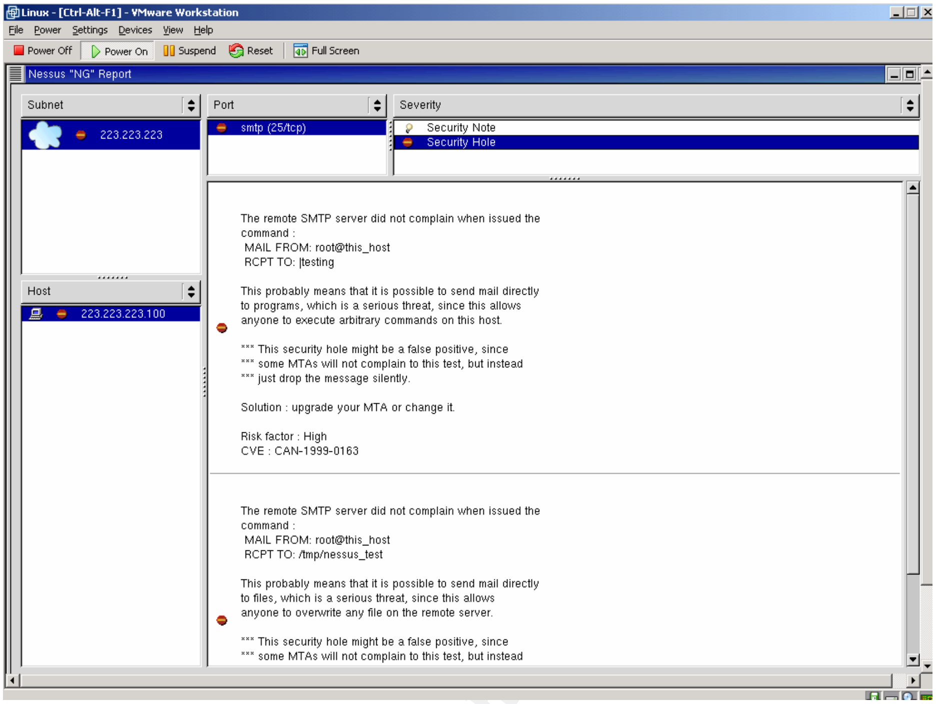
```
smtpscan was not able to reliably identify this server. It might be:
Microsoft Exchange 2000
The fingerprint differs from these known signatures on 2 point(s)

If you known precisely what it is, please send this fingerprint
to the Nessus team :
:503:500:501:250:501:250:250:500:500:500:500:500:500:250:500
```

Diagram 3 – Mail Server Detection

This shows that Nessus has correctly identified the Mail server as an Exchange server.

The next screenshots show that Nessus has been able to run additional tests and has identified a range of serious vulnerabilities on this Mail server.


Linux - [Ctrl-Alt-F1] - VMware Workstation
File Power Settings Devices View Help
Power Off Power On Suspend Reset Full Screen
Nessus "NG" Report
Subnet
223.223.223
Host
223.223.223.100
Port
smtp (25/tcp)
Severity
Security Note
Security Hole

The remote SMTP server did not complain when issued the command :
 MAIL FROM: root@this_host
 RCPT TO: |testing

This probably means that it is possible to send mail directly to programs, which is a serious threat, since this allows anyone to execute arbitrary commands on this host.

*** This security hole might be a false positive, since
*** some MTAs will not complain to this test, but instead
*** just drop the message silently.

Solution : upgrade your MTA or change it.

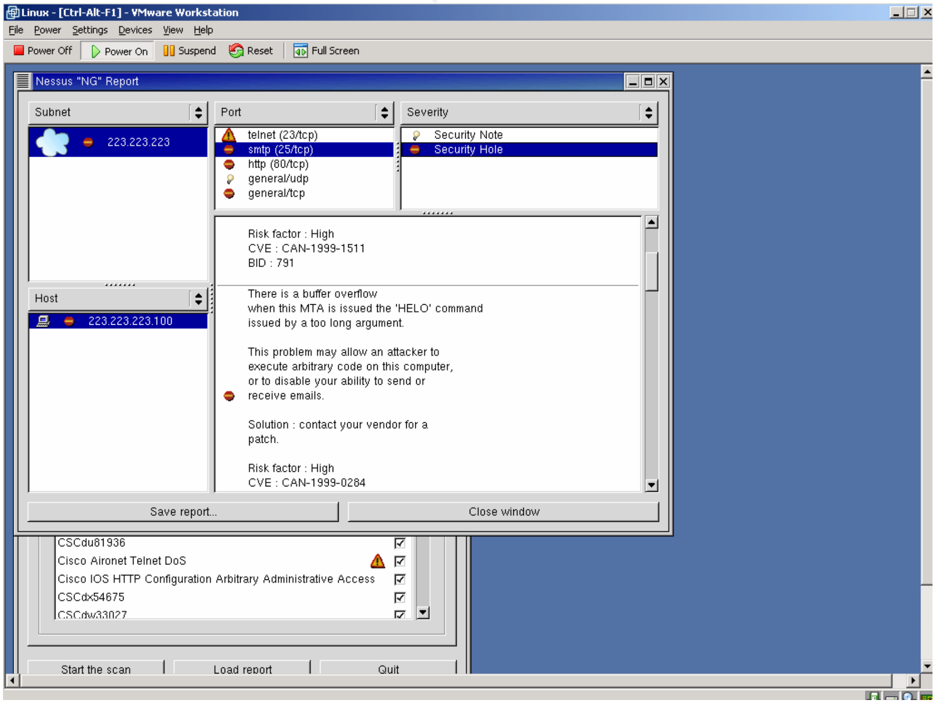
Risk factor : High
CVE : CAN-1999-0163

The remote SMTP server did not complain when issued the command :
 MAIL FROM: root@this_host
 RCPT TO: /tmp/nessus_test

This probably means that it is possible to send mail directly to files, which is a serious threat, since this allows anyone to overwrite any file on the remote server.

*** This security hole might be a false positive, since
*** some MTAs will not complain to this test, but instead

Diagram 4 – Mail Server Vulnerabilities.



Risk factor : High
CVE : CAN-1999-1511
BID : 791

There is a buffer overflow when this MTA is issued the 'HELO' command issued by a too long argument.

This problem may allow an attacker to execute arbitrary code on this computer, or to disable your ability to send or receive emails.

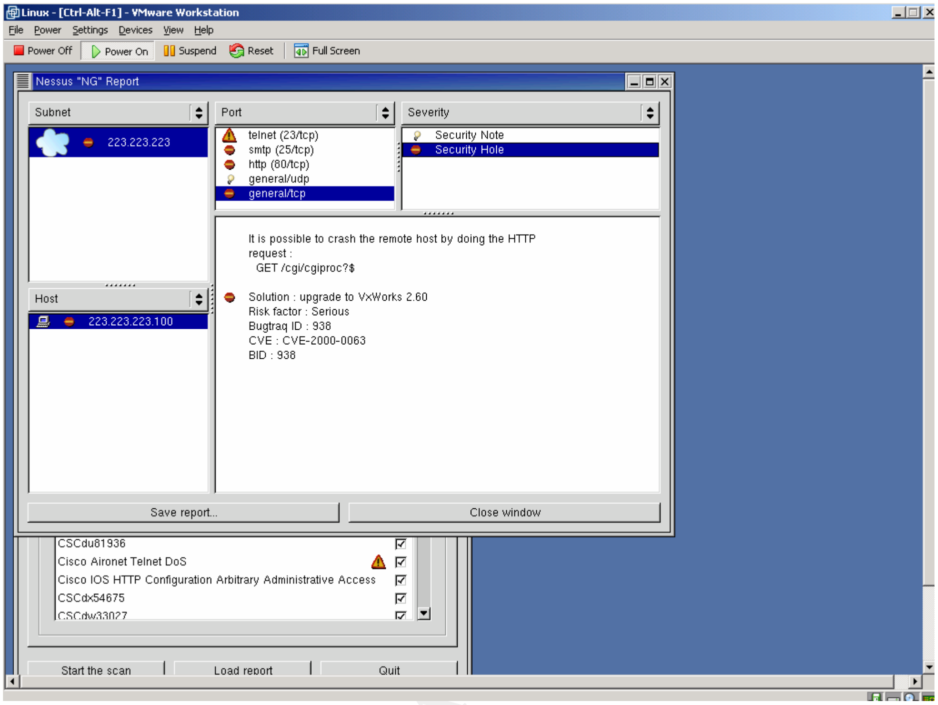
Solution : contact your vendor for a patch.

Risk factor : High
CVE : CAN-1999-0284

Diagram 5 – Buffer Overflow Vulnerabilities.

The above screenshots show that this Mail server has some serious vulnerabilities!

The next screenshots identify some possible vulnerabilities on a web server.


Linux - [Ctrl-Alt-F1] - VMware Workstation
File Power Settings Devices View Help
Power Off Power On Suspend Reset Full Screen
Nessus "NG" Report
Subnet
Port
Severity
Host
Save report...
Close window
Start the scan
Load report
Quit

Subnet: 223.223.223

Host: 223.223.223.100

Port: telnet (23/tcp), smtp (25/tcp), http (80/tcp), general/udp, general/tcp

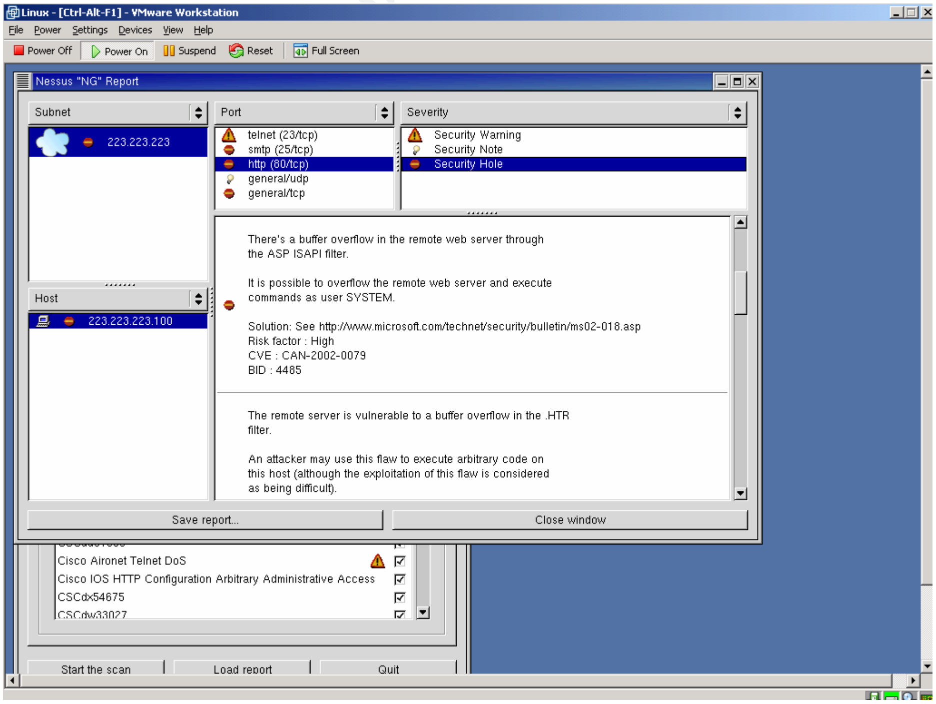
Severity: Security Note, Security Hole

It is possible to crash the remote host by doing the HTTP request :
GET /cgi/cgiproc?$

Solution : upgrade to VxWorks 2.60
Risk factor : Serious
Bugtraq ID : 938
CVE : CVE-2000-0063
BID : 938

CSCdu81936 ☑
Cisco Aironet Telnet DoS ☑
Cisco IOS HTTP Configuration Arbitrary Administrative Access ☑
CSCdx54675 ☑
CSCdw33027 ☑

Diagram 6 – HTTP vulnerabilities.


Linux - [Ctrl-Alt-F1] - VMware Workstation
File Power Settings Devices View Help
Power Off Power On Suspend Reset Full Screen
Nessus "NG" Report
Subnet
Port
Severity
Host
Save report...
Close window
Start the scan
Load report
Quit

Subnet: 223.223.223

Host: 223.223.223.100

Port: telnet (23/tcp), smtp (25/tcp), http (80/tcp), general/udp, general/tcp

Severity: Security Warning, Security Note, Security Hole

There's a buffer overflow in the remote web server through the ASP ISAPI filter.

It is possible to overflow the remote web server and execute commands as user SYSTEM.

Solution: See http://www.microsoft.com/technet/security/bulletin/ms02-018.asp
Risk factor : High
CVE : CAN-2002-0079
BID : 4485

The remote server is vulnerable to a buffer overflow in the .HTR filter.

An attacker may use this flaw to execute arbitrary code on this host (although the exploitation of this flaw is considered as being difficult).

Cisco Aironet Telnet DoS ☑
Cisco IOS HTTP Configuration Arbitrary Administrative Access ☑
CSCdx54675 ☑
CSCdw33027 ☑

Diagram 7 – Web Server Vulnerabilities.

It is quite clear from the Nessus scans that we need to patch these servers immediately! As we have mentioned previously, had we had our patch management procedures in place, we would hopefully not have had any production servers in such as exposed state. Patch management is yet another layer of our defence system.

4.17 SANS Top 20 List

The SANs top 20 is a list of the 20 most common vulnerabilities that your security perimeter should protect against. Some of these SANs top20 list includes services such as NTP, DNS and Syslog, which we need to permit though our firewall. However we have ensured that these ports are only permitted either from required hosts or to specific hosts. For example TCP 53 (DNS zone transfers) is only permitted between our external DNS server and our ISP secondary DNS servers.

As suggested by SANS at http://www.sans.org/top20.htm, we have verified that our firewall blocks these vulnerabilities.

Our guiding principle has been to deny everything and then only permit specific traffic flows.

4.18 Audit Summary

We have spent considerable time planning our perimeter security and firewall rules. Our testing has been successful in that we have not found any unexpected holes in our firewall. We did however find some duplicate rules that would never be matched, due to a preceding rule, so we have re-ordered and updated the access-lists.

Some of the NMAP scans (FIN, Null and Xmas Tree) were all reported as OPEN by NMAP, but the PIX did actually drop the packets.

We have found that the firewall only permits expected traffic and except for a few minor access list ordering problems, the firewall rule base met our expectations.

We have scanned the firewall thoroughly with NMAP and did not find any unexpected open ports. Testing has verified that the rule base works correctly.

We encountered a bizarre problem with the PIX crashing and reloading when we attempted to initiate a connection to the PGP key server. The workaround is to disable the fixup for ILS or to upgrade to PIX software version 6.3. This is detailed in Cisco bug id's CSCdx78331 and CSCdx73007. We do not want to upgrade to version 6.3 as it is not yet EAL4 certified, so we will disable the fixup protocol for ILS on port 389.

Finally we used the Nessus tool to run some additional tests and although these were run against some unpatched servers, this demonstrates that a firewall can only provide part of the required security protection and re-iterates how important it is to have a defence in depth approach.

5. Design Under Fire

The security design chosen for review is that of Susan Delaney, GCFW Analyst number 0412. The original practical can be found at: http://www.giac.org/practical/GCFW/Susan_Delaney_GCFW.pdf
A diagram of the network under review is shown below.

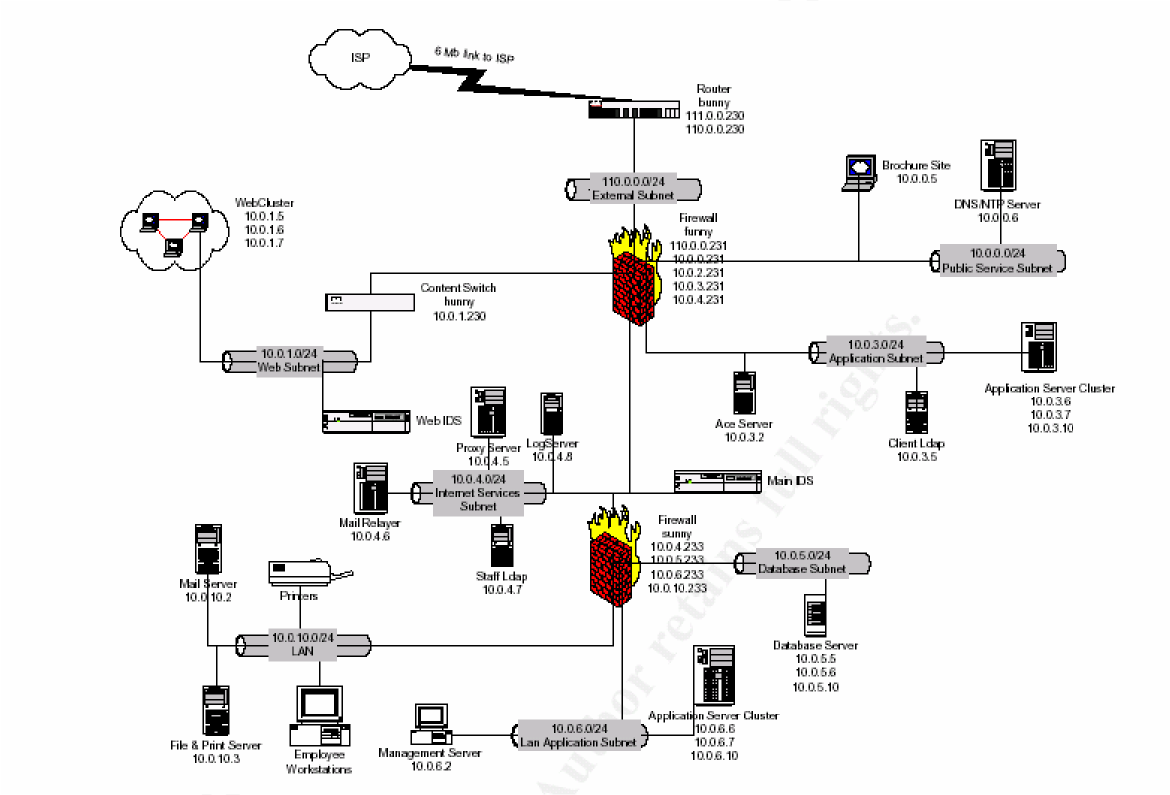


Diagram 8 - Diagram taken from Susan Delaney's GCFW Practical

5.1 Attack against the Firewall

The external firewall in use is Checkpoint Firewall-1/VPN NH FP3 with Hotfix-2. This is running on a Sun Solaris 2.8 platform. A search for vulnerabilities has identified the following possibilities:

5.1.1 Syslog Daemon Vulnerability

Syslog daemon vulnerability is identified at:
http://www.secunia.com/advisories/8371/
An extract form this advisory is shown below:

> *"**Description**:*
> *Two vulnerabilities have been discovered in the syslog daemon included in some versions of Check Point FireWall-1.*

> *One vulnerability allows people to crash the syslog daemon by sending large amounts of data to the syslog service. It has been discussed whether this vulnerability could be exploited to execute arbitrary code, but it has not been proven.*
>
> *The other allows malicious users to inject malicious characters such as console escape sequences. This could be dangerous depending on the utility used to read the log files.*
>
> *The syslog service is not enabled by default.*
>
> *The vulnerability has been confirmed in the following versions:*
>
> ** Check Point FW-1 NG FP3*
> ** Check Point FW-1 NG FP2*
> ** Check Point FW-1 4.1 SP6*
> ***Solution**:*
> *Secunia recommends that you use a dedicated host for remote logging. It is an unnecessary risk to run it on your firewall - no matter how convenient it may be.*
>
> *An update (HF2) fixing the first issue is available via SmartUpdate or from:*
> *http://www.checkpoint.com/techsupport/ng/fp3_hotfix.html*
>
> *The other issue concerning injection of malicious characters has not been fixed. We recommend that you use a different syslog server or filter the log files with some other tool before viewing them.*
> ***Reported by / credits**:*
> *Dr. Peter Bieringer"*

Although this is fixed with FP3 HF2, (http://www.checkpoint.com/techsupport/ng/fp3_hotfix.html) it may be that the firewall has not actually been upgraded with this hotfix yet, so it is still worth attempting this exploit.
Some information and details on how to perform this exploit can be found at:

http://lists.netsys.com/pipermail/full-disclosure/2003-March/004589.html

As this attack involves sending valid syslog messages, the chances are the IDS systems would not flag this as an attack.
Unfortunately (for us) the external routers will filter any syslog traffic, so even if the firewall has not been upgraded this exploit cannot be launched against the external firewall.

5.1.2 IKE vulnerability

Another potential vulnerability is with IKE. As the external firewall is used as the VPN termination point, this exploit may yield some success. This vulnerability is detailed at:
http://www.securityfocus.com/archive/1/291340

http://www.securityfocus.com/archive/1/290202/2002-09-01/2002-09-07/0
Susan is using the external Checkpoint firewall as the remote access VPN gateway, terminating all IPSEC VPN sessions on the firewall. Even though this vulnerability report is in reference to SecureRemote and Susan has chosen to deploy SecureClient, the vulnerability is actually within IKE, so it still possible that this vulnerability can be exploited.

The vulnerability can be exploited if IKE is used in aggressive mode with shared secrets. It is possible to identify valid usernames with a dictionary attack and then brute force their passwords. Susan is using shared secrets, so it is possible that this still exists in FP3.

Firewall-1 does not enforce a minimum password length, so good password management is required to ensure that strong passwords are used. Another weakness with Firewall-1 is that there is no account lockout feature – i.e. lock out an account after a number of failed attempts, so repeated brute force password attempts can be made against one particular userid. Once a valid username and password had been discovered an IPSEC / ESP session could be established to the firewall with full VPN privileges to perform further reconnaissance and launch additional attacks or collect information.

One solution to this problem would be to deploy a PKI and use digital certificates for authentication.

If excessive IKE invalid user authentication attempts can be sent to the firewall, this can cause a huge CPIU spike within Firewall-1 as the authentication process if very CPU intensive. Whilst not compromising the firewall, it is a form of DOS that can be used to attack the firewall.

There are not any IDS on the outside of the firewall in Susan's design, but this would definitely be noticed in the firewall logs! There are currently not any methods with Checkpoint to rate limit the number of IKE authentication attempts, but CAR could be configured on the external router to limit IKE traffic as an additional protection mechanism.

Additional vulnerability information can be found at the following vulnerability research sites

- Security focus – http://www.online.securityfocus.com
- Mitres common vulnerabilities and exposures – http://www.cve.mitre.org

- CERT – http://www.cert.org

5.2 Denial of Service Attack

A denial of service attack can take many forms, such as; flooding a link with unwanted traffic that consumes all available bandwidth, targeting a specific server and attempting to starve that server of available resources, such as a SYN flood attack, running exploits against vulnerable services or compromising someone's DNS records with a bogus IP Address, but the objective is to make a service (or entire site) unavailable.

Here we will outline how a distributed denial of service attack (DDOS) can be launched from fifty compromised cable modem/DSL systems.

The first thing to do is map the target site for potential DDOS victims. For this exercise we have been able to examine the external router access lists to determine which traffic will actually get through. From examining these access control lists we can see that RFC1918 and IANA reserved source IP Addresses will be blocked, so we cannot use these as our spoofed address ranges. The router will however permit any icmp packet-too-big messages, any traffic without the SYN flag set and any SYN packets for TCP ports 80, 443, 25, 18231, 18234 and any UDP traffic on ports 53, 264 and 500, plus any IKE connections. This gives us quite a wide range of potential DDOS targets.

In reality we wouldn't have had such easy visibility of the access control lists and would have used a tool such as NMAP to find open addresses and ports, but we run the risk of being picked up by the IDS. To circumvent this and to mask our NMAP scan, some other compromised hosts can be used to launch a decoy scan with the aim of making the IDS log files very large to the point of rendering them useless. We can use another host to run our 'stealth' NMAP scan to find both open ports perform OS fingerprinting.

Of course we could just simply use our zombies to flood the link from the ISP to GIAC. The available bandwidth between GIAC and the ISP is likely 6Mbps. With 50 compromised hosts running at a full 128Kbps, we could potentially flood the ISP link (50 x 128Kbps = 6.4Mbps). If the links is fully utilised, then this will impact genuine customer, partner and supplier traffic to the GIAC website. Users will experience extremely slow connections, connections will drop out and they will not be able to connect. GIAC relies on the Internet connection to run its business, so this type of attack would effectively shut down GIAC's business operations.

Assuming we have managed to compromise 50 cable modem / DSL systems, we now need to decide with DDOS tool we are going to use. The most popular DDOS tools are Trinoo, TFN, TFN2K and Stacheldraht. The diagram below shows the topology for a DDOS attack using the Stacheldraht attack tool.

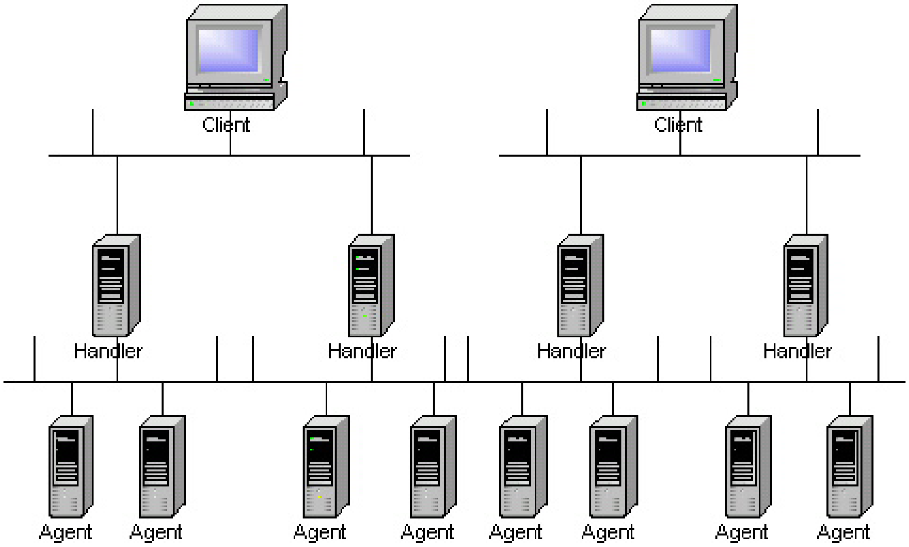


Diagram 9 – Stacheldraht Topology. Source www.cisco.com

In summary, the client is the person who orchestrates the attack, a handler is a compromised host that runs special control code and the agent is another compromised host controlled by a handler. The attack packets to the target are initiated from the agents.

For more information please refer to the following:

- http://www.cisco.com/en/US/customer/tech/tk583/tk385/technologies_white_paper09186a0080174a5b.shtml
- http://www.staff.washington.edu/dittrich/misc/trinoo.analysis
- http://www.staff.washington.edu/dittrich/misc/tfn.analysis
- http://staff.washington.edu/dittrich/misc/stacheldraht.analysis

For our attack we have chosen to use the TFN2K attack tool, which consists of a master and agent component. The master is used to pass commands to the agents who reside on our compromised hosts. First I have to compile the code on my Linux system. Before compiling you need to edit the src/Makefile and uncomment the options for your operating system. There are various options in src/config.h that can be changed from default. TFN2K uses encrypted communication with the agents to pass commands and command packets can be interleaved with decoy packets. This makes TFN2K extremely difficult to detect.

To commence a co-ordinated DDOS attack I need to list all of fifty compromised hosts in a file, that is used by the TFN2K client to contact the agents.

For my attack I will start with a UDP flood attack toward the GIAC site, consuming all of their bandwidth, but I can optionally run further attacks using some of the TFN2K options.

TFN2K has many options including being able to spoof the source address and initiate other attacks such as SYN flood, ICMP, SMURF and combined attacks, which are known as MIX attacks. These use a combination of UDP, SYN and ICMP packets. I will not post any TFN2K commands here, but needless to say to launch the attack is fairly trivial.

An excellent analysis of TFN2K can be found at

- http://security.royans.net/info/posts/bugtraq_ddos2.shtml

5.2.1 Countermeasures

Prevention of a DDOS attack is virtually impossible and responding to such an attack is very difficult and time consuming and will almost certainly require co-operation with the ISP. Whilst GIAC has some defence measures in place, such as the SYN Defender feature of Checkpoints SmartDefense, this will only assist in functions such as only passing on a TCP connection once the TCP 3-way handshake has completed. GIAC use a Cisco content switch which has some built in DOS prevention capabilities. It is similar to SYN Defender, in that it waits for the 3-WAY handshake between itself and the source host to complete before opening a connection to the web server. The CSS will wait sixteen seconds for the SYN-ACK and then will drop the flow. Additionally if it receives eight consecutive SYNs from the same source address it will just drop any further SYN packets with the same initial sequence number form that source[13]. This is a valuable tool to prevent the web server form being starved of connection resources. However in an attack aimed at flooding your bandwidth you need to prevent the traffic getting to you in the first place and this is where GIAC will require assistance form the ISP. In the case of a flooding attack and if the ISP is running Cisco routers you can request them to enable a Cisco IOS feature called Committed Access Rate (CAR). This is a bandwidth rate-limiting tool and can used to restrict the amount of bandwidth a particular IP address or protocol can consume.

Another countermeasure is to run a free tool called 'ZombieZapper' which can be found at:

- http://razor.bindview.com/tools/ZombieZapper_form.shtml.

[13] http://www.cisco.com/en/US/customer/products/hw/contnetw/ps789/products_white_paper09186a00800921a6.shtml

5.3 Attack Plan to Compromise an internal system through the perimeter system

The first thing we need to do is identify some information about the site and our starting point will be nslookup.

We will first attempt to identify the GIAC name server and attempt a zone transfer.

```
C:\>nslookup
Default Server:  ns1.optus.net.au
Address:  202.139.83.3
> set type=ns
> www.giac.com
> www.cisco.com
Server:  ns1.optus.net.au
Address:  202.139.83.3
giac.com
      primary name server = dns.giac.com
      responsible mail addr = postmaster.giac.com
      serial  = 4041561
      refresh = 7200 (2 hours)
      retry   = 1800 (30 mins)
      expire  = 864000 (10 days)
      default TTL = 86400 (1 day)
>
> server dns.gia.com
Default Server:  dns.giac.com
Address:  10.10.10.100 (dummy address shown here)
```

We now attempt a zone transfer

```
> ls -d giac.com
[dns.giac.com]
*** Can't list domain giac.com: Query refused
>
```

This tells us that zone transfers have been restricted, so not much luck with being able use DNS to obtain any internal network and host information.

Next we query the InterNIC databases.[14] We search the RIPE 'WHOIS' database at http://www.ripe.net/db/whois/whois.html and run a search against the GIAC nameserver ip address.

Some sample output from two whois queries reveals the following about GIAC:

```
netnum:     xx.xxx.xxx.0 - xxx.xxx.xxx.255
netname:     GIAC-IE
descr:      GIAC Enterprises
country:     IE
admin-c:     GIAC-RIPE
tech-c:      GIAC-RIPE
mnt-by:      GIAC-MNT
changed:     nobody@ripe.net 20030509
status:      ALLOCATED PORTABLE
source:      RIPE


Search results for: ! NET-XX-XX-1-0-1
OrgName:    GIAC
OrgID:      GIAC-1
Address:    35 Industrial Way
City:       Dublin
Country:    IE
NetRange:   xxx.xxx.xxx.0 - xxx.xxx.xxx.255
CIDR:       xxx.xxx.xxx.0/24
NetName:    GIAC-1
NetHandle:  NET-xx-xx-x-0-1
Parent:     NET-xx-0-0-0-1
NetType:    Reassigned
NameServer: DNS.GIAC.COM
TechPhone:  +353-123-123-1234
TechEmail:  dnstech@giac.com
```

[14] SANS GCFW Training – Perimeter Protection Day 6.

Susan has done a good job here to ensure that only minimal information is available, but we can still obtain some useful details such as a company contact telephone number and a site postal address.

Next we run an NMAP stealth scan to identify which hosts and ports are open. This will provide us with information on any potentially vulnerable services. We could also try some banner grabbing, i.e. telnet to a host on a particular port and see if we can glean any information on software versions and OS platforms.

We next run a search at http://uptime.netcraft.com and find that although the servers seem to have been patched recently, the previous change was over 6 months ago. This could just be that the netcraft site does not have correct information, but it could mean that GIAC are very slack at patching servers. We will continue to monitor this; as if a new vulnerability is detected we may have a window of opportunity before an updated patch is applied.

5.3.1 Selected Target

It is quite clear that the perimeter security is very tight, so our options are to run further attack tools and try and exploit some of the servers and hope to gain remote access to these or to use some good old fashioned social engineering techniques and attempt to compromise an internal system.

5.3.2 Process to compromise the target

One way to compromise an internal host is to first identify a valid email address. These can be found by trawling through various mailing lists such as Yahoo, as these do not generally remove the original email header. SMTP headers can reveal some valuable information in addition to the user, such as mail server versions, MIME version and details of mail relays / sweepers.[15]

Once we have the user information we can send them a flashy email with a hyperlink to a web site that we own. The hyperlink can be used to download an activeX object with an embedded executable and launch this on the user's workstation, such as netcat or cryptcat which can then be utilised to enable access to the compromised host.

More information can be found at:

- http://www.trojanforge.net/showthread.php?s=&threadid=8015
- http://www.packetstormsecurity.nl/0005-exploits/silent.delivery.txt

[15] SANs GCFW Training – Perimeter Protection Day 6.

5.3.3 Countermeasures

To protect against server OS and application vulnerabilities it is essential that a patch management system is in place. This will ensure that change management procedures are in place to control the testing and updating of vendor patches as soon as they become available.

Social engineering (especially via telephone calls) is still a wide open doorway into many organisations and it essential that an adequate security policy is developed to the requirement of each organisation. This security policy should be made available to all employees and regular security awareness programmes should be conducted to make all staff aware of such threats.

To prevent the silent delivery of executables, i.e. without the user knowing, ensure that browsers are running a recent version with high security enabled. Again, user eduction is a vital component of your defence strategy to mitigate this kind of risk.

Appendix AReference Sources

1. OS Hardening Tools - YASSP - http://www.yassp.org and Bastille http://www.bastille-linux.org.
2. Router Security Configuration Guide - NSA-Router Security Configuration Guide - http://nsa1.www.conxion.com/cisco/
3. APNIC - http://www.apnic.net/db/AS.html
4. PIX Firewall Configuration - http://www.cisco.com/en/US/customer/products/sw/secursw/ps2120/products_command_reference_chapter09186a008010423d.html
5. PIX Device Manager (PDM) - http://www.cisco.com/en/US/customer/products/sw/netmgtsw/ps2032/products_installation_guide_chapter09186a00800e3826.html
6. DNS BIND software - http://www.isc.org/products/BIND/
7. Sendmail software and patches - http://www.sendmail.com/support/download/patch_page.shtml
8. Advanced Encryption Standard (AES) - http://www.esat.kuleuven.ac.be/~rijmen/rijndael/
9. NMAP - http://www.insecure.org/nmap
10. Susan Delaney GCFW Practical http://www.giac.org/practical/GCFW/Susan_Delaney_GCFW.pdf
11. Distributed Denial of Service (DDOS) Attacks - http://www.cisco.com/en/US/customer/tech/tk583/tk385/technologies_white_paper09186a0080174a5b.shtml
12. TFN2K – http://www.packetstorm.deceptions.ord/disributed/
13. Trinoo - http://www.staff.washington.edu/dittrich/misc/trinoo.analysis
14. TFN - http://www.staff.washington.edu/dittrich/misc/tfn.analysis
15. Stacheldraht - http://staff.washington.edu/dittrich/misc/stacheldraht.analysis
16. DDOS - http://security.royans.net/info/posts/bugtraq_ddos2.shtml
17. Zombie Zapper - http://razor.bindview.com/tools/ZombieZapper_form.shtml
18. Web Site Information Tool - Netcraft http://uptime.netcraft.com
19. Silent Delivery - http://www.trojanforge.net/showthread.php?s=&threadid=8015
20. Silent Delivery - http://www.packetstormsecurity.nl/0005-exploits/silent.delivery.txt

21. Inside Network Perimeter Security – Stephen Northcutt, Lenny Zeltser, Scott Winters, Karen Kent Frederick, Ronald W. Ritchey. ISBN 0-7357-1232-8

22. Cisco Content Switches Web-site Security and Denial-of-Service Protection - http://www.cisco.com/en/US/customer/products/hw/contnetw/ps789/products_white_paper09186a00800921a6.shtml

Appendix B Device Summary

Device Name	Hardware	Software	Description
3725-1	Cisco 3725	IOS 12.2.15T2	Border Router
3725-2	Cisco 3725	IOS 12.2.15T2	Border Router
2950-1	Cisco 2950	IOS 12.1.13.EA1c	DMZ Switch
2950-2	Cisco 2950	IOS 12.1.13.EA1c	DMZ Switch
SYDFW01	Cisco PIX 525 Firewall	Unrestricted Licence version 6.2.(2)	External Firewall
SYDFW02	Cisco PIX 525 Firewall	Failover Licence version 6.2.2	External Firewall
2950-3	Cisco 2950	IOS 12.1.13.EA1c	VPN screened subnet switch
2950-4	Cisco 2950	IOS 12.1.13.EA1c	VPN screened subnet switch
VPN01	Cisco 3015	3.6 vpn3000-3.6.7.F-9.bin	Cisco VPN Concentrator
2950-5	Cisco 2950	IOS 12.1.13.EA1c	Partner / supplier server screened subnet switch
2950-6	Cisco 2950	IOS 12.1.13.EA1c	Partner / supplier server screened subnet switch
CSS-11503-1	Cisco Content Switch	WebNS 7.20.0.3	Intelligent content switch
CSS-11503-2	Cisco Content Switch	WebNS 7.20.0.3	Intelligent content switch
CTE1400-1	Cisco Transformation Engine	CTEServer 2.7	Web Content Transformation Appliance
2950-7	Cisco 2950	IOS 12.1.13.EA1c	Web screened subnet switch
2950-8	Cisco 2950	IOS 12.1.13.EA1c	Web screened subnet switch
2950-9	Cisco 2950	IOS 12.1.13.EA1c	Transit screened subnet switch
2950-10	Cisco 2950	IOS 12.1.13.EA1c	Transit screened subnet switch
CHKP-1	Nokia IP 530 Appliance	Nokia IPSO Checkpoint Firewall-1 NG FP3	Checkpoint Firewall

Device Name	Hardware	Software	Description
		Hotfix 2	
CHKP-2	Nokia IP 530 Appliance	Nokia IPSO Checkpoint Firewall-1 NG FP3 Hotfix 2	Checkpoint Firewall
2950-11	Cisco 2950	IOS 12.1.13.EA1c	Application screened subnet switch
2950-12	Cisco 2950	IOS 12.1.13.EA1c	Application screened subnet switch
2950-13	Cisco 2950	IOS 12.1.13.EA1c	Database screened subnet switch
2950-14	Cisco 2950	IOS 12.1.13.EA1c	Database screened subnet switch
2950-15	Cisco 2950	IOS 12.1.13.EA1c	Management / Logging screened subnet switch
2950-16	Cisco 2950	IOS 12.1.13.EA1c	Management / Logging screened subnet switch
2503-1	Cisco 2503	IOS 12.1.20	Sink Hole Router
4506-1	Cisco 6509	SUP IV IOS 12.1.19E	Internal Layer 3 Switch
4506-1	Cisco 6509	SUP IV IOS 12.1.19E	Internal Layer 3 Switch
Web-1	SunFire 280R	Solaris 8 SunONE Web Server 6 service pack 4	Web Server
Web-2	SunFire 280R	Solaris 8 SunONE Web Server 6 service pack 4	Web Server
Web-3	SunFire 280R	Solaris 8 SunONE Web Server 6 service pack 4	Web Server
Web-4	SunFire 280R	Solaris 8 SunONE Web Server 6 service pack 4	Web Server

Device Name	Hardware	Software	Description
FTP-1	Compaq	Linux RedHat 7.3	Supplier FTP Server
Oracle-Rep	SunFire V120	Solaris 8 Oracle 9i (9.2.02)	Replica Oracle Server
E-DNS	SunFire V120	Solaris 8 BIND V9.2.2	Extenal DNS
I-DNS	SunFire V120	Solaris 8 BIND V9.2.2	Internal DNS
E-SMTP	SunFire V120	Solaris 8 Sendmail x.x.x	External Mail Server
I-MAIL	Compaq	MS Windows 2000 Server MS Exchange 2000	Internal email Server
App-1	SunFire V480	Solaris 8 IBM Websphere 5	Websphere Application server
App-2	SunFire V480	Solaris 8 IBM Websphere 5	Websphere Application server
LDAP-P	SunFire V120	Solaris 8 Sun LCO Directory Proxy Server 5.9 (LDAP Proxy)	LDAP Proxy Server
Oracle	SunFire V120	Solaris 8 Oracle 9i (9.2.02)	Oracle Database Server
LDAP	SunFire V120	Solaris 8 OpenLDAP 2.0.25	LDAP Server
Proxy	SunFire V120	Solaris 8 Netscape iPlanet Proxy Server 3.6 SP2	Web Proxy server
IDS-1	Compaq	SNORT	Web IDS server
IDS-2	Compaq	SNORT	Supplier / Partner IDS server
IDS-3	Compaq	SNORT	Appication IDS server

Device Name	Hardware	Software	Description
IDS-4	Compaq	SNORT	Database IDS server
IDS-5	Compaq	SNORT	Mail / DNS IDS server
IDS-6	Compaq	SNORT	Management IDS server
IDS-7	Compaq	SNORT	VPN IDS server
Syslog-1	Compaq	Syslog-ng	Syslog Server
Syslog-1	Compaq	Syslog-ng	Syslog Server
NTP-1	Compaq	XNTPD	NTP Server
NTP-2	Compaq	XNTPD	NTP Server

Appendix C GIAC-1 External Router Configuration

```
!
version 12.2
no service pad
service tcp-keepalives-in
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone
no service password-encryption
no service dhcp
!
hostname GIAC-1
!
logging buffered 16384 debugging
no logging console
enable secret 5 $1$ZijH$sRUQVP0kIveQNBi.Mst8E1
!
username user1 password 0 password
aaa new-model
!
aaa authentication login GIAC local-case
aaa session-id common
ip subnet-zero
no ip source-route
ip cef
!
ip tftp source-interface loopback0
no ip domain lookup
!
no ip bootp server
!
interface Loopback0
 description ------ Loopback Interface for BGP and OSPF ------
 ip address 10.64.8.1 255.255.255.255
!
interface FastEthernet0/0
 description ------ Link to ISPX ------
 ip address 223.223.254.253 255.255.255.252
 ip access-group 199 in
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 speed 10
 duplex half
 no cdp enable
!
interface FastEthernet0/1
 description ------ DMZ LAN ------
 ip access-group 198 in
 ip address 223.223.223.1 255.255.255.0
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 speed 100
 full-duplex
 no cdp enable
 standby 1 ip 223.223.223.3
 standby 1 timers 2 5
 standby 1 priority 120
 standby 1 preempt
 standby 1 track FastEthernet0/0 30
!
interface Ethernet1/0
 description ------ Crossover to GIAC-2 ------
 bandwidth 10000
 ip address 10.64.8.253 255.255.255.252
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 ip ospf authentication
 ip ospf authentication-key 7 0506071D2D1C40
 ip ospf priority 2
 full-duplex
 no cdp enable
!
```

```
router ospf 100
 log-adjacency-changes
 redistribute connected subnets route-map Into_OSPF
 network 10.0.0.0 0.255.255.255 area 0.0.0.0
 default-metric 100
!
router bgp 65500
 no synchronization
 bgp log-neighbor-changes
 network 223.223.223.0
 neighbor 10.64.8.2 remote-as 65500
 neighbor 10.64.8.2 description IBGP to GIAC-2
 neighbor 10.64.8.2 update-source Loopback0
 neighbor 10.64.8.2 timers 5 15
 neighbor 10.64.8.2 next-hop-self
 neighbor 10.64.8.2 default-originate
 neighbor 10.64.8.2 soft-reconfiguration inbound
 neighbor 223.223.254.254 remote-as 64551
 neighbor 223.223.254.254 description EBGP to ISPX1
 neighbor 223.223.254.254 prefix-list ISPX1_Default in
 neighbor 223.223.254.254 distribute-list 1 out
 neighbor 223.223.254.254 route-map default_route_preference in
 neighbor 223.223.254.254 filter-list 2 out
 no auto-summary
!
ip classless
no ip http server
ip as-path access-list 2 permit ^65500$
ip as-path access-list 2 permit ^$
ip as-path access-list 2 deny .*
ip ospf name-lookup
!
!
ip prefix-list ISPX1_Default seq 5 permit 0.0.0.0/0
!
logging history notifications
logging trap errors
logging facility local5
logging source-interface Loopback0
logging 192.168.11.11
logging 192.168.11.12
access-list 10 remark Distribute FA0/1 LAN into OSPF on Crossover Link
access-list 10 permit 223.223.223.0 0.0.0.255
access-list 100 remark VTY Access List
access-list 100 permit tcp host 192.168.11.50 host 10.64.8.1 eq 23 log-input
access-list 100 deny ip any any log-input
access-list 198 remark To internet Filter
access-list 198 deny   ip 127.0.0.0 0.255.255.255 any log
access-list 198 deny   ip 0.0.0.0 0.255.255.255 any log
access-list 198 deny   ip 1.0.0.0 0.255.255.255 any log
access-list 198 deny   ip 2.0.0.0 0.255.255.255 any log
access-list 198 deny   ip 10.0.0.0 0.255.255.255 any log
access-list 198 deny   ip 23.0.0.0 0.255.255.255 any log
access-list 198 deny   ip 31.0.0.0 0.255.255.255 any log
access-list 198 deny   ip 67.0.0.0 0.255.255.255 any log
access-list 198 deny   ip 68.0.0.0 3.255.255.255 any log
access-list 198 deny   ip 72.0.0.0 3.255.255.255 any log
access-list 198 deny   ip 80.0.0.0 15.255.255.255 any log
access-list 198 deny   ip 96.0.0.0 15.255.255.255 any log
access-list 198 deny   ip 112.0.0.0 3.255.255.255 any log
access-list 198 deny   ip 126.0.0.0 1.255.255.255 any log
access-list 198 deny   ip 169.254.0.0 0.0.255.255 any log
access-list 198 deny   ip 172.16.0.0 0.15.255.255 any log
access-list 198 deny   ip 191.255.0.0 0.0.255.255 any log
access-list 198 deny   ip 192.0.2.0 0.0.0.255 any log
access-list 198 deny   ip 192.168.0.0 0.0.255.255 any log
access-list 198 deny   ip 198.18.0.0 0.0.255.255 any log
access-list 198 deny   ip 201.0.0.0 0.255.255.255 any log
access-list 198 deny   ip 223.255.255.0 0.0.0.255 any log
access-list 198 deny   ip 224.0.0.0 31.255.255.255 any log
access-list 198 deny   tcp any any range 135 139 log
access-list 198 deny   tcp any any eq 445 log
access-list 198 deny   icmp any any time-exceeded
access-list 198 deny   icmp any any host-unreachable
access-list 198 deny   icmp any any echo-reply
access-list 198 permit ip host 223.223.223.7 any
```

```
access-list 198 permit udp host 223.223.223.75 any eq isakmp
access-list 198 permit esp host 223.223.223.75 any
access-list 198 permit tcp host 223.223.223.75 eq 10000 any
access-list 198 permit tcp host 223.223.223.75 host 1.1.1.1 eq www
access-list 198 deny   ip any any log
access-list 199 remark From internet Filter
access-list 199 deny   ip 127.0.0.0 0.255.255.255 any log
access-list 199 deny   ip 0.0.0.0 0.255.255.255 any log
access-list 199 deny   ip 1.0.0.0 0.255.255.255 any log
access-list 199 deny   ip 2.0.0.0 0.255.255.255 any log
access-list 199 deny   ip 10.0.0.0 0.255.255.255 any log
access-list 199 deny   ip 23.0.0.0 0.255.255.255 any log
access-list 199 deny   ip 31.0.0.0 0.255.255.255 any log
access-list 199 deny   ip 67.0.0.0 0.255.255.255 any log
access-list 199 deny   ip 68.0.0.0 3.255.255.255 any log
access-list 199 deny   ip 72.0.0.0 3.255.255.255 any log
access-list 199 deny   ip 80.0.0.0 15.255.255.255 any log
access-list 199 deny   ip 96.0.0.0 15.255.255.255 any log
access-list 199 deny   ip 112.0.0.0 3.255.255.255 any log
access-list 199 deny   ip 126.0.0.0 1.255.255.255 any log
access-list 199 deny   ip 169.254.0.0 0.0.255.255 any log
access-list 199 deny   ip 172.16.0.0 0.15.255.255 any log
access-list 199 deny   ip 191.255.0.0 0.0.255.255 any log
access-list 199 deny   ip 192.0.2.0 0.0.0.255 any log
access-list 199 deny   ip 192.168.0.0 0.0.255.255 any log
access-list 199 deny   ip 198.18.0.0 0.0.255.255 any log
access-list 199 deny   ip 201.0.0.0 0.255.255.255 any log
access-list 199 deny   ip 222.255.255.0 0.0.0.255 any log
access-list 199 deny   ip 223.255.255.0 0.0.0.255 any log
access-list 199 deny   ip 223.0.0.0 0.255.255.255 any log
access-list 199 deny   ip 224.0.0.0 31.255.255.255 any log
access-list 199 permit tcp any host 223.223.223.100 eq www
access-list 199 permit tcp any host 223.223.223.50 eq domain
access-list 199 permit udp any host 223.223.223.50 eq domain
access-list 199 permit udp any host 223.223.223.12 eq 25
access-list 199 permit tcp any host 223.223.223.100 eq 443
access-list 199 permit udp any host 223.223.223.75 eq isakmp
access-list 199 permit esp any host 223.223.223.75
```

```
access-list 199 permit tcp any host 223.223.223.75 eq 10000
access-list 199 permit tcp host 223.223.254.254 host 223.223.254.253 eq bgp
access-list 199 permit tcp any any established
access-list 199 permit icmp any 223.223.223.0 0.0.0.255 echo-reply
access-list 199 permit icmp any 223.223.223.0 0.0.0.255 host-unreachable
access-list 199 permit icmp any 223.223.223.0 0.0.0.255 net-unreachable
access-list 199 permit icmp any 223.223.223.0 0.0.0.255 packet-too-big
access-list 199 deny   ip any any log
no cdp run
!
route-map default_route_preference permit 10
 match ip address prefix-list ISPX1_Default
 set weight 100
!
route-map Into_OSPF permit 10
 match ip address 10
!
banner exec ^CCCCC
******************************************************************
******************************************************************
                         WARNING
Access to this system is for authorised users and for authorised
purposes only.
Unauthorised access or use is a serious breach of security policies.
For staff this may involve disciplinary action up to and including
dismissal, it may also be a criminal or civil offence.
If you or your intended use are not authorised do not proceed to log on
to this system.
******************************************************************
***************************************************************************^C
banner motd ^CCCCC
******************************************************************
******************************************************************
This network, information about its components and information systems
within it are CONFIDENTIAL. Access to, or use of, this network by
unauthorised people (including subsidiary companies and their personnel)
or for any other unauthorised purpose is STRICTLY PROHIBITED.
This Router records and logs user IP addresses.
```

```
*****************************************************************
************************************************************************* ^C
!
line con 0
 session-timeout 5
 exec-timeout 5 0
 login authentication GIAC
 transport output none
line aux 0
 no exec
line vty 0 4
 access-class 100 in
 session-timeout 5
 exec-timeout 5 0
 login authentication GIAC
!
ntp authentication-key 40 md5 105D0C1A171206 7
ntp authenticate
ntp source Loopback0
ntp server 172.25.1.101
ntp server 172.25.2.102
!
end
```

Appendix DPIX Configuration

```
PIX Version 6.2(2)
nameif ethernet0 outside security0
nameif ethernet1 inside security100
nameif ethernet2 partsup security50
nameif ethernet3 web security50
nameif ethernet4 vpn security20
nameif ethernet5 sync security90
enable password 2KFQnbNIdI.2KYOU encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
hostname SYDFW01
domain-name giac.com.au
clock timezone AST 11
fixup protocol ftp 21
fixup protocol http 80
fixup protocol h323 h225 1720
fixup protocol h323 ras 1718-1719
fixup protocol ils 389
fixup protocol rsh 514
fixup protocol rtsp 554
fixup protocol smtp 25
fixup protocol sqlnet 1521
fixup protocol sip 5060
fixup protocol skinny 2000
no names
object-group network vpn_network_devices
  network-object 192.168.3.50 255.255.255.255
  network-object 192.168.3.51 255.255.255.255
  network-object 192.168.3.3 255.255.255.255
  network-object 192.168.3.4 255.255.255.255
object-group network part_sup_network_devices
  network-object 192.168.4.3 255.255.255.255
  network-object 192.168.4.4 255.255.255.255
object-group network web_network_devices
  network-object 192.168.5.3 255.255.255.255
  network-object 192.168.5.4 255.255.255.255
  network-object 192.168.5.23 255.255.255.255
  network-object 192.168.5.24 255.255.255.255
object-group network external_network_devices
  network-object 10.64.8.1 255.255.255.255
  network-object 10.64.8.2 255.255.255.255
  network-object 223.223.223.5 255.255.255.255
  network-object 223.223.223.6 255.255.255.255
object-group network GIAC_vpn_pools
  network-object 10.10.1.0 255.255.255.0
  network-object 10.10.2.0 255.255.255.0
  network-object 10.10.3.0 255.255.255.0
object-group network internal_hosts
  network-object 172.25.100.0 255.255.255.0
  network-object 172.25.200.0 255.255.255.0
object-group network app_servers
  network-object 192.168.9.10 255.255.255.255
  network-object 192.168.9.11 255.255.255.255
object-group network spoofed_networks
  network-object 0.0.0.0 255.0.0.0
  network-object 1.0.0.0 255.0.0.0
  network-object 2.0.0.0 255.0.0.0
  network-object 10.0.0.0 255.0.0.0
  network-object 23.0.0.0 255.0.0.0
  network-object 31.0.0.0 255.0.0.0
  network-object 67.0.0.0 255.0.0.0
  network-object 68.0.0.0 252.0.0.0
  network-object 72.0.0.0 252.0.0.0
  network-object 80.0.0.0 240.0.0.0
  network-object 96.0.0.0 240.0.0.0
  network-object 112.0.0.0 252.0.0.0
  network-object 126.0.0.0 254.0.0.0
  network-object 127.0.0.0 255.0.0.0
  network-object 169.254.0.0 255.255.0.0
  network-object 172.16.0.0 255.240.0.0
  network-object 191.255.0.0 255.255.0.0
network-object 192.168.0.0 255.255.0.0
  network-object 198.18.0.0 255.255.0.0
  network-object 201.0.0.0 255.0.0.0
  network-object 222.255.255.0 255.255.255.0
  network-object 224.0.0.0 224.0.0.0
object-group network ntp_servers
  network-object 223.223.223.23 255.255.255.255
  network-object 223.223.223.24 255.255.255.255
object-group network syslog_servers
  network-object 223.223.223.21 255.255.255.255
  network-object 223.223.223.22 255.255.255.255
object-group network isp_dns_servers
  network-object 202.139.83.3 255.255.255.255
  network-object 192.65.90.202 255.255.255.255
object-group service webservices tcp
  description inbound HTTP and SSL
  port-object eq www
  port-object eq https
object-group service web_management tcp
  description web servers management
  port-object eq 8081
  port-object eq 9173
  port-object eq 16187
object-group network part_sup_vpn_pools
network-object 10.10.10.0 255.255.255.0
  network-object 10.10.20.0 255.255.255.0
object-group network appservers
  network-object 192.168.9.10 255.255.255.255
  network-object 192.168.9.11 255.255.255.255
object-group service appserver_ports tcp
  description websphere ports
  port-object eq 9080
  port-object eq 9443
object-group network web_servers
  network-object 192.168.5.101 255.255.255.255
  network-object 192.168.5.102 255.255.255.255
  network-object 192.168.5.103 255.255.255.255
  network-object 192.168.5.104 255.255.255.255
object-group network inside_address_syslog_servers
  network-object 192.168.11.11 255.255.255.255
  network-object 192.168.11.12 255.255.255.255
object-group network syslog_servers_real
```

```
  network-object 192.168.11.11 255.255.255.255
  network-object 192.168.11.12 255.255.255.255
object-group network ntp_servers_real
  network-object 172.25.1.101 255.255.255.255
  network-object 172.25.1.102 255.255.255.255
object-group network inside_address_ntp_servers
 network-object 172.25.1.101 255.255.255.255
  network-object 172.25.2.102 255.255.255.255
access-list outside_access_in deny ip object-group spoofed_networks any
access-list outside_access_in permit tcp any host 223.223.223.100 object-group webservices
access-list outside_access_in permit udp any host 223.223.223.50 eq domain
access-list outside_access_in permit tcp any host 223.223.223.12 eq smtp
access-list outside_access_in permit udp object-group external_network_devices object-group syslog_servers eq syslog
access-list outside_access_in permit icmp any 223.223.223.0 255.255.255.0 echo-reply
access-list outside_access_in permit icmp any 223.223.223.0 255.255.255.0 unreachable
access-list outside_access_in permit icmp any 223.223.223.0 255.255.255.0 time-exceeded
access-list outside_access_in permit tcp object-group external_network_devices object-group ntp_servers eq 123
access-list outside_access_in permit tcp object-group isp_dns_servers host 223.223.223.50 eq domain
access-list outside_access_in permit tcp any host 223.223.223.21 object-group webservices
access-list outside_access_in permit udp object-group external_network_devices host 223.223.223.13 eq tftp
access-list inside_outbound_nat0_acl permit ip host 192.168.11.50 object-group vpn_network_devices
access-list inside_outbound_nat0_acl permit ip host 192.168.11.50 object-group part_sup_network_devices
access-list inside_outbound_nat0_acl permit ip host 192.168.11.50 object-group web_network_devices
access-list inside_outbound_nat0_acl permit ip object-group internal_hosts object-group web_servers
access-list inside_outbound_nat0_acl permit ip object-group internal_hosts host 192.168.5.111

access-list inside_outbound_nat0_acl permit ip host 172.25.1.50 host 192.168.5.50
access-list inside_outbound_nat0_acl permit ip object-group internal_hosts host 192.168.4.10
```

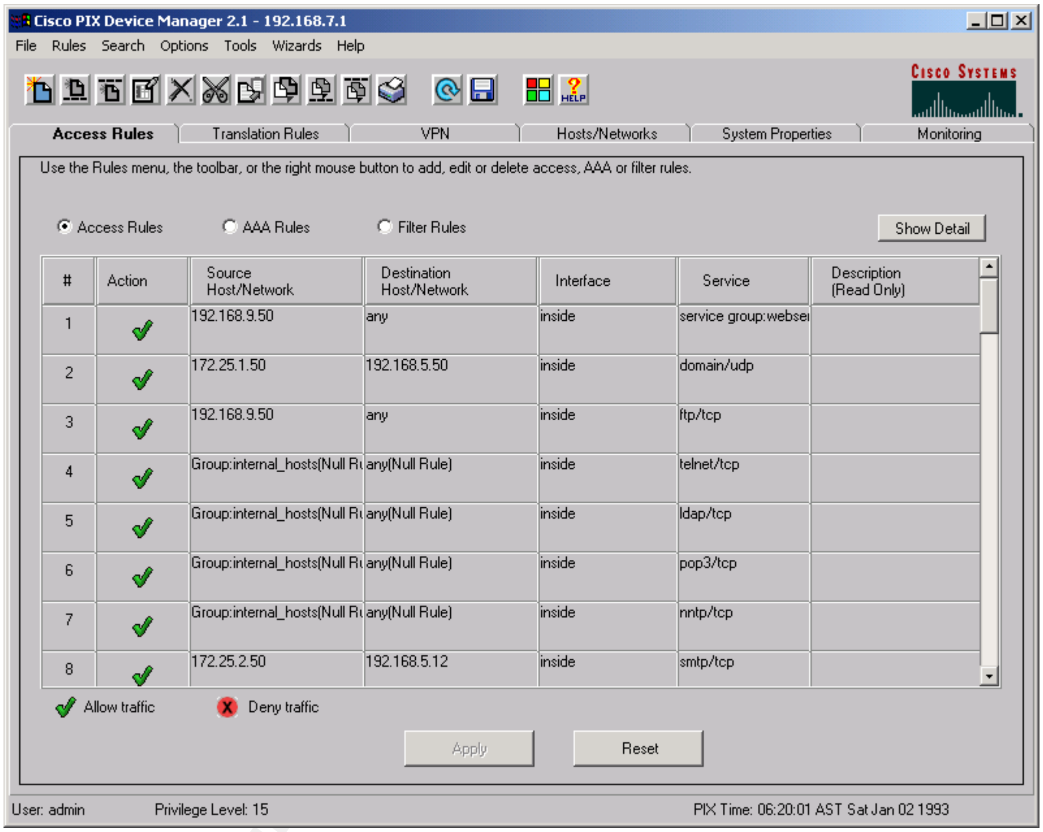
```
access-list inside_outbound_nat0_acl permit ip object-group internal_hosts host 192.168.5.100
access-list vpn_access_in permit tcp object-group GIAC_vpn_pools host 223.223.223.100 object-group webservices
access-list vpn_access_in permit udp object-group GIAC_vpn_pools host 172.25.1.50 eq domain
access-list vpn_access_in permit udp object-group vpn_network_devices object-group inside_address_syslog_servers eq syslog
access-list vpn_access_in permit tcp object-group GIAC_vpn_pools host 172.25.2.50 eq smtp
access-list vpn_access_in permit tcp object-group part_sup_vpn_pools host 192.168.4.10 eq sqlnet
access-list vpn_access_in permit tcp object-group part_sup_vpn_pools host 192.168.4.11 eq ftp
access-list vpn_access_in permit tcp 10.10.2.0 255.255.255.0 host 192.168.4.10 eq sqlnet
access-list vpn_access_in permit tcp 10.10.2.0 255.255.255.0 host 192.168.4.11 eq ftp
access-list vpn_access_in permit tcp 10.10.3.0 255.255.255.0 host 192.168.4.10
access-list vpn_access_in permit tcp 10.10.3.0 255.255.255.0 host 192.168.4.11
access-list vpn_access_in permit tcp object-group vpn_network_devices object-group inside_address_ntp_servers eq 123
access-list vpn_access_in permit udp object-group vpn_network_devices host 192.168.11.13 eq tftp
access-list vpn_access_in permit tcp object-group GIAC_vpn_pools any object-group webservices
access-list web_access_in permit tcp host 192.168.5.200 object-group app_servers object-group appserver_ports
access-list web_access_in permit tcp host 192.168.5.200 host 192.168.9.100 eq ldap
access-list web_access_in permit udp object-group web_network_devices object-group inside_address_syslog_servers eq syslog
access-list web_access_in permit udp host 192.168.5.50 any eq domain
access-list web_access_in permit tcp host 192.168.5.12 any eq smtp
access-list web_access_in permit tcp host 192.168.5.50 object-group isp_dns_servers eq domain
access-list web_access_in permit tcp object-group web_network_devices object-group inside_address_ntp_servers eq 123
access-list web_access_in permit udp object-group web_network_devices host 192.168.11.13 eq tftp
access-list partsup_access_in permit udp object-group part_sup_network_devices object-group inside_address_syslog_servers eq syslog
access-list partsup_access_in permit tcp object-group part_sup_network_devices object-group inside_address_ntp_servers eq 123
access-list partsup_access_in permit udp object-group part_sup_network_devices host 192.168.11.13 eq tftp
```

```
access-list sync_access_in deny ip any any
access-list inside_access_in permit tcp host 192.168.9.50 any object-group webservices
access-list inside_access_in permit udp host 172.25.1.50 host 192.168.5.50 eq domain
access-list inside_access_in permit tcp host 192.168.9.50 any eq ftp
access-list inside_access_in permit tcp object-group internal_hosts any eq telnet
access-list inside_access_in permit tcp object-group internal_hosts any eq ldap
access-list inside_access_in permit tcp object-group internal_hosts any eq pop3
access-list inside_access_in permit tcp object-group internal_hosts any eq nntp
access-list inside_access_in permit tcp host 172.25.2.50 host 192.168.5.12 eq smtp
access-list inside_access_in permit tcp object-group internal_hosts host 192.168.4.10 eq sqlnet
access-list inside_access_in permit tcp object-group internal_hosts object-group web_servers object-group web_management
access-list inside_access_in permit tcp object-group internal_hosts host 192.168.5.111 eq 9001
access-list inside_access_in permit tcp host 192.168.11.50 any eq ssh
access-list inside_access_in deny icmp any any time-exceeded
access-list inside_access_in deny icmp any any unreachable
access-list inside_access_in deny icmp any any echo-reply
access-list inside_access_in permit icmp object-group internal_hosts any
access-list inside_access_in permit icmp host 192.168.11.50 any
pager lines 24
logging on
logging timestamp
logging standby
logging monitor informational
logging buffered debugging
logging trap informational
logging facility 23
logging host inside 192.168.11.11
logging host inside 192.168.11.12
interface ethernet0 100full
interface ethernet1 100full
interface ethernet2 100full
interface ethernet3 100full
interface ethernet4 100full
interface ethernet5 100full
mtu outside 1500
mtu inside 1500
mtu partsup 1500
mtu web 1500
mtu vpn 1500
mtu sync 1500
ip address outside 223.223.223.7 255.255.255.0
ip address inside 192.168.7.1 255.255.255.0
ip address partsup 192.168.4.1 255.255.255.0
ip address web 192.168.5.1 255.255.255.0
ip address vpn 192.168.3.1 255.255.255.0
ip address sync 192.168.2.1 255.255.255.0
ip verify reverse-path interface outside
ip verify reverse-path interface inside
ip verify reverse-path interface partsup
ip verify reverse-path interface web
ip verify reverse-path interface vpn
ip verify reverse-path interface sync
ip audit info action alarm
ip audit attack action alarm
no failover
failover timeout 0:00:00
failover poll 3
failover ip address outside 223.223.223.8
failover ip address inside 192.168.7.2
failover ip address partsup 192.168.4.2
failover ip address web 192.168.5.2
failover ip address vpn 192.168.3.2
failover ip address sync 192.168.2.2
failover link sync
pdm location 172.25.0.0 255.255.0.0 inside
pdm location 192.168.7.10 255.255.255.255 inside
pdm location 192.168.9.0 255.255.255.0 inside
pdm location 192.168.11.11 255.255.255.255 inside
pdm location 192.168.11.12 255.255.255.255 inside
pdm location 192.168.11.0 255.255.255.0 inside
pdm location 10.10.1.0 255.255.255.0 vpn
pdm location 10.10.2.0 255.255.255.0 vpn
pdm location 10.10.3.0 255.255.255.0 vpn
pdm location 10.10.10.0 255.255.255.0 vpn
pdm location 10.10.20.0 255.255.255.0 vpn
pdm location 172.25.1.101 255.255.255.255 inside
pdm location 172.25.1.102 255.255.255.255 inside
pdm location 192.168.5.12 255.255.255.255 web
pdm location 192.168.5.50 255.255.255.255 web
pdm location 192.168.5.100 255.255.255.255 web
pdm location 192.168.5.3 255.255.255.255 web
pdm location 192.168.5.4 255.255.255.255 web
pdm location 192.168.5.23 255.255.255.255 web
pdm location 192.168.5.24 255.255.255.255 web
pdm location 192.168.4.3 255.255.255.255 partsup
pdm location 192.168.4.4 255.255.255.255 partsup
pdm location 192.168.3.3 255.255.255.255 vpn
pdm location 192.168.3.4 255.255.255.255 vpn
pdm location 192.168.3.50 255.255.255.255 vpn
pdm location 192.168.3.51 255.255.255.255 vpn
pdm location 10.10.1.0 255.255.255.0 web
pdm location 172.25.0.0 255.255.0.0 web
pdm location 192.168.4.10 255.255.255.255 web
pdm location 192.168.4.11 255.255.255.255 web
pdm location 172.25.1.50 255.255.255.255 inside
pdm location 172.25.2.50 255.255.255.255 inside
pdm location 192.168.9.50 255.255.255.255 inside
pdm location 192.168.11.50 255.255.255.255 inside
pdm location 192.168.5.200 255.255.255.255 web
pdm location 192.168.11.11 255.255.255.255 vpn
pdm location 192.168.11.12 255.255.255.255 vpn
pdm location 0.0.0.0 255.255.255.0 outside
pdm location 1.0.0.0 255.255.255.0 outside
pdm location 2.0.0.0 255.255.255.0 outside
pdm location 10.0.0.0 255.255.255.0 outside
pdm location 10.64.8.1 255.255.255.255 outside
pdm location 10.64.8.2 255.255.255.255 outside
pdm location 23.0.0.0 255.0.0.0 outside
pdm location 31.0.0.0 255.255.255.0 outside
pdm location 67.0.0.0 255.255.255.0 outside
pdm location 68.0.0.0 255.255.252.0 outside
pdm location 72.0.0.0 255.255.252.0 outside
pdm location 80.0.0.0 255.255.240.0 outside
pdm location 96.0.0.0 255.255.240.0 outside
pdm location 112.0.0.0 255.255.252.0 outside
pdm location 126.0.0.0 255.255.254.0 outside
pdm location 127.0.0.0 255.255.255.0 outside
pdm location 169.254.0.0 255.255.0.0 outside
pdm location 172.16.0.0 255.255.240.0 outside
pdm location 191.255.0.0 255.255.0.0 outside
pdm location 192.65.90.202 255.255.255.255 outside
pdm location 192.168.9.10 255.255.255.255 outside
pdm location 192.168.9.11 255.255.255.255 outside
pdm location 192.168.9.100 255.255.255.255 outside
pdm location 192.168.0.0 255.255.0.0 outside
pdm location 198.18.0.0 255.255.0.0 outside
pdm location 201.0.0.0 255.255.255.0 outside
pdm location 202.139.83.3 255.255.255.255 outside
pdm location 222.255.255.0 255.255.255.0 outside
pdm location 223.223.223.5 255.255.255.255 outside
pdm location 223.223.223.6 255.255.255.255 outside
pdm location 223.223.223.21 255.255.255.255 outside
pdm location 223.223.223.23 255.255.255.255 outside
pdm location 224.0.0.0 255.255.224.0 outside
pdm location 192.168.3.3 255.255.255.255 inside
pdm group web_network_devices web
pdm group part_sup_network_devices partsup
pdm group vpn_network_devices vpn
pdm group web_dmz web
pdm group vpn_pools vpn
pdm group part_sup_dmz web
pdm group internal_hosts web
pdm group GIAC_vpn_pools vpn
pdm group syslog_servers_real inside
pdm group ntp_servers_real inside
pdm group inside_address_syslog_servers vpn
pdm group spoofed_networks outside
pdm group external_network_devices outside
pdm group syslog_servers outside reference syslog_servers_real
pdm group ntp_servers outside reference ntp_servers_real
pdm group isp_dns_servers outside
pdm group app_servers outside
pdm history enable
arp timeout 14400
global (outside) 1 interface
global (outside) 2 223.223.223.250
nat (inside) 0 access-list inside_outbound_nat0_acl
nat (inside) 1 192.168.9.50 255.255.255.255 0 0
nat (inside) 1 172.25.100.0 255.255.255.0 0 0
nat (inside) 1 172.25.200.0 255.255.255.0 0 0
nat (web) 1 192.168.5.12 255.255.255.255 0 0
nat (web) 1 192.168.5.50 255.255.255.255 0 0
nat (vpn) 2 10.10.1.0 255.255.255.0 0 0
nat (vpn) 2 10.10.2.0 255.255.255.0 0 0
nat (vpn) 2 10.10.3.0 255.255.255.0 0 0
static (inside,outside) 223.223.223.21 192.168.11.11 netmask 255.255.255.255 0 0
static (inside,outside) 223.223.223.22 192.168.11.12 netmask 255.255.255.255 0 0
static (inside,outside) 223.223.223.23 172.25.1.101 netmask 255.255.255.255 0 0
static (inside,outside) 223.223.223.24 172.25.1.102 netmask 255.255.255.255 0 0
static (web,outside) 223.223.223.12 192.168.5.12 netmask 255.255.255.255 0 0
static (web,outside) 223.223.223.100 192.168.5.100 netmask 255.255.255.255 0 0
static (web,vpn) 223.223.223.100 192.168.5.100 netmask 255.255.255.255 0 0
static (inside,vpn) 192.168.11.11 192.168.11.11 netmask 255.255.255.255 0 0
static (inside,vpn) 172.25.1.50 172.25.1.50 netmask 255.255.255.255 0 0
static (inside,vpn) 172.25.1.101 172.25.1.101 netmask 255.255.255.255 0 0
static (inside,vpn) 172.25.2.102 172.25.2.102 netmask 255.255.255.255 0 0
static (inside,vpn) 172.25.2.50 172.25.2.50 netmask 255.255.255.255 0 0
static (partsup,vpn) 192.168.4.10 192.168.4.10 netmask 255.255.255.255 0 0
static (partsup,vpn) 192.168.4.11 192.168.4.11 netmask 255.255.255.255 0 0
static (inside,vpn) 192.168.11.12 192.168.11.12 netmask 255.255.255.255 0 0
static (inside,partsup) 172.25.1.101 172.25.1.101 netmask 255.255.255.255 0 0
static (inside,partsup) 172.25.2.102 172.25.2.102 netmask 255.255.255.255 0 0
static (inside,partsup) 192.168.11.11 192.168.11.11 netmask 255.255.255.255 0 0
static (inside,partsup) 192.168.11.12 192.168.11.12 netmask 255.255.255.255 0 0
static (inside,web) 172.25.1.101 172.25.1.101 netmask 255.255.255.255 0 0
static (inside,web) 172.25.2.102 172.25.2.102 netmask 255.255.255.255 0 0
static (inside,web) 192.168.11.11 192.168.11.11 netmask 255.255.255.255 0 0
static (inside,web) 192.168.11.12 192.168.11.12 netmask 255.255.255.255 0 0
static (inside,web) 172.25.2.50 172.25.2.50 netmask 255.255.255.255 0 0
static (inside,web) 192.168.9.100 192.168.9.100 netmask 255.255.255.255 0 0
static (inside,web) 192.168.9.10 192.168.9.10 netmask 255.255.255.255 0 0
static (inside,web) 192.168.9.11 192.168.9.11 netmask 255.255.255.255 0 0
static (inside,outside) 223.223.223.13 192.168.11.13 netmask 255.255.255.255 0 0
static (inside,web) 192.168.11.13 192.168.11.13 netmask 255.255.255.255 0 0
static (inside,vpn) 192.168.11.13 192.168.11.13 netmask 255.255.255.255 0 0
static (inside,partsup) 192.168.11.13 192.168.11.13 netmask 255.255.255.255 0 0
access-group outside_access_in in interface outside
access-group inside_access_in in interface inside
access-group partsup_access_in in interface partsup
access-group web_access_in in interface web
access-group vpn_access_in in interface vpn
access-group sync_access_in in interface sync
route outside 0.0.0.0 0.0.0.0 223.223.223.3 1
route vpn 10.10.1.0 255.255.255.0 192.168.3.50 1
route vpn 10.10.2.0 255.255.255.0 192.168.3.50 1
route vpn 10.10.3.0 255.255.255.0 192.168.3.50 1
route vpn 10.10.10.0 255.255.255.0 192.168.3.50 1
route vpn 10.10.20.0 255.255.255.0 192.168.3.50 1
route inside 172.25.0.0 255.255.0.0 192.168.7.5 1
route inside 192.168.9.0 255.255.255.0 192.168.7.5 1
route inside 192.168.11.0 255.255.255.0 192.168.7.5 1
timeout xlate 3:00:00
timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00 rpc 0:10:00 h323 0:05:00 sip 0:30:00 sip_media 0:02:00
timeout uauth 0:05:00 absolute
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
aaa-server LOCAL protocol local
aaa authentication enable console LOCAL
aaa authentication http console LOCAL
aaa authentication serial console LOCAL
aaa authentication ssh console LOCAL
aaa authorization command LOCAL
ntp server 172.25.1.101 source inside
ntp server 172.25.2.102 source inside
```

```
http server enable
http 192.168.11.50 255.255.255.255 inside
no snmp-server location
no snmp-server contact
snmp-server community public
no snmp-server enable traps
floodguard enable
no sysopt route dnat
telnet timeout 5
ssh 192.168.11.50 255.255.255.255 inside
ssh timeout 5
username monitor password 28OsuAeHLWtpOHWa encrypted privilege 3
username admin password 7KKG/zg/Wo8c.YfN encrypted privilege 15
privilege show level 0 command version
privilege show level 0 command curpriv
privilege show level 3 command pdm
privilege show level 3 command blocks
privilege show level 3 command ssh
privilege configure level 3 command who
privilege show level 3 command isakmp
privilege show level 3 command ipsec
privilege show level 3 command vpdn
privilege show level 3 command local-host
privilege show level 3 command interface
privilege show level 3 command ip
privilege configure level 3 command ping
privilege show level 3 command uauth
privilege configure level 5 mode enable command configure
privilege show level 5 command running-config
privilege show level 5 command privilege
privilege show level 5 command clock
privilege show level 5 command ntp
privilege show level 5 mode configure command logging
terminal width 80
Cryptochecksum:12606208e6121bfe27f1fb44f9fb002f
: end
SYDFW01#
```

Appendix E PDM Sample Screenshots

The following are sample screenshots of Cisco's PIX Device Manaer (PDM). This is web based configuration and monitoring.


Cisco PIX Device Manager 2.1 - 192.168.7.1
File Rules Search Options Tools Wizards Help
Cisco Systems
Access Rules
Translation Rules
VPN
Hosts/Networks
System Properties
Monitoring

Use the Rules menu, the toolbar, or the right mouse button to add, edit or delete access, AAA or filter rules.

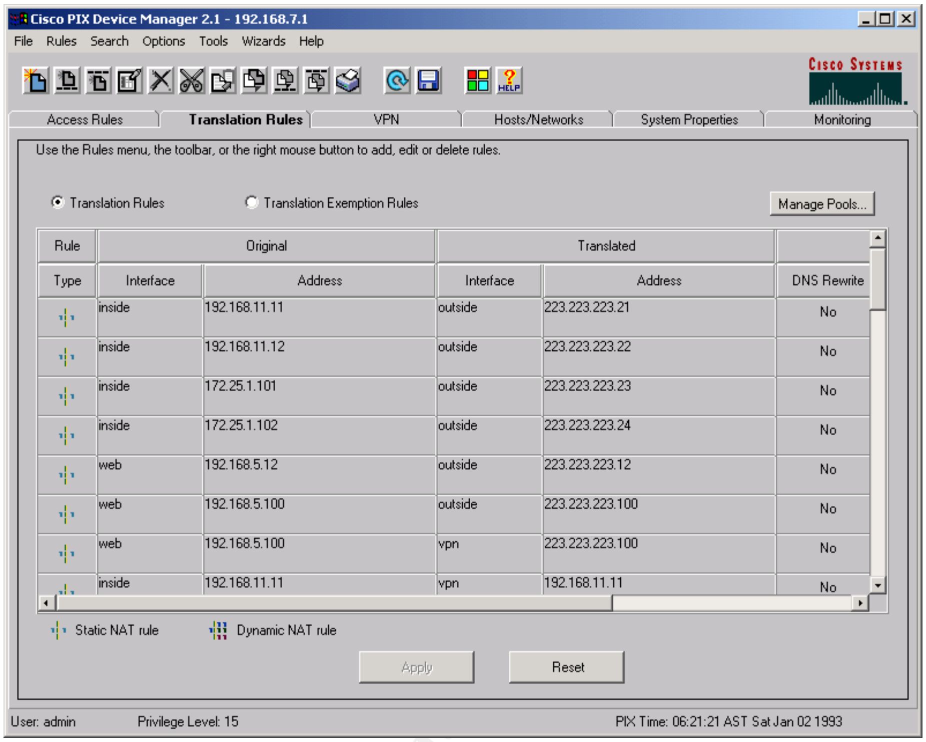
● Access Rules ○ AAA Rules ○ Filter Rules

#	Action	Source Host/Network	Destination Host/Network	Interface	Service	Description (Read Only)
1	✓	192.168.9.50	any	inside	service group:websei	
2	✓	172.25.1.50	192.168.5.50	inside	domain/udp	
3	✓	192.168.9.50	any	inside	ftp/tcp	
4	✓	Group:internal_hosts(Null Ri	any(Null Rule)	inside	telnet/tcp	
5	✓	Group:internal_hosts(Null Ri	any(Null Rule)	inside	ldap/tcp	
6	✓	Group:internal_hosts(Null Ri	any(Null Rule)	inside	pop3/tcp	
7	✓	Group:internal_hosts(Null Ri	any(Null Rule)	inside	nntp/tcp	
8	✓	172.25.2.50	192.168.5.12	inside	smtp/tcp	

✓ Allow traffic ☒ Deny traffic

User: admin Privilege Level: 15 PIX Time: 06:20:01 AST Sat Jan 02 1993

Diagram 10 – PIX Access Rules



Use the Rules menu, the toolbar, or the right mouse button to add, edit or delete rules.

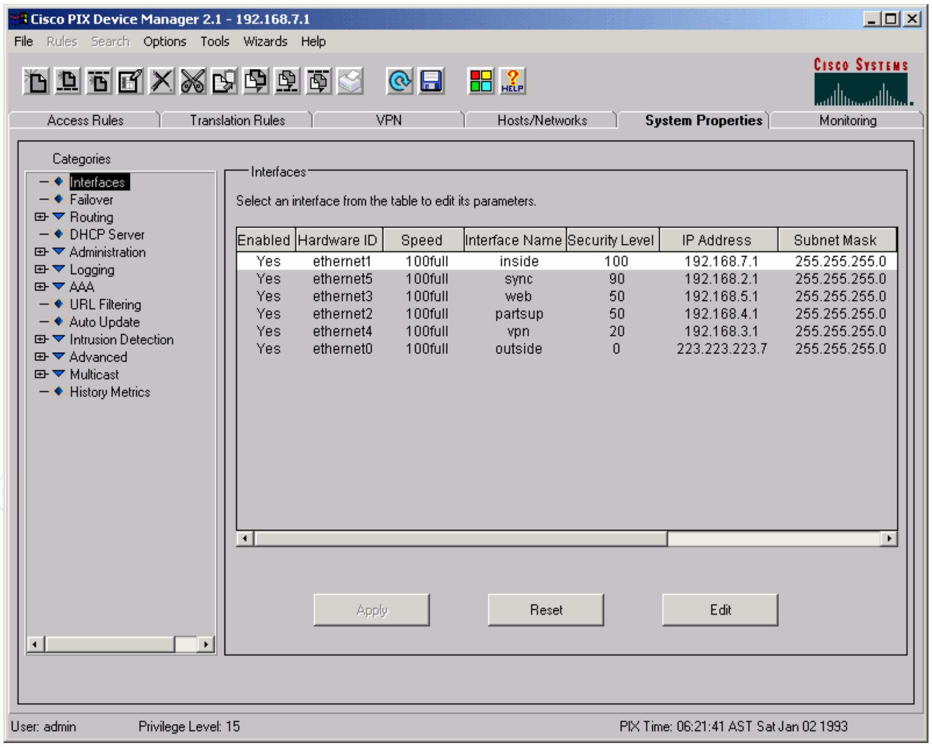
● Translation Rules ○ Translation Exemption Rules

Rule	Original		Translated		
Type	Interface	Address	Interface	Address	DNS Rewrite
✛	inside	192.168.11.11	outside	223.223.223.21	No
✛	inside	192.168.11.12	outside	223.223.223.22	No
✛	inside	172.25.1.101	outside	223.223.223.23	No
✛	inside	172.25.1.102	outside	223.223.223.24	No
✛	web	192.168.5.12	outside	223.223.223.12	No
✛	web	192.168.5.100	outside	223.223.223.100	No
✛	web	192.168.5.100	vpn	223.223.223.100	No
✛	inside	192.168.11.11	vpn	192.168.11.11	No

✛ Static NAT rule ⁞ Dynamic NAT rule

User: admin Privilege Level: 15 PIX Time: 06:21:21 AST Sat Jan 02 1993

Diagram 11 – PIX Translations Rules



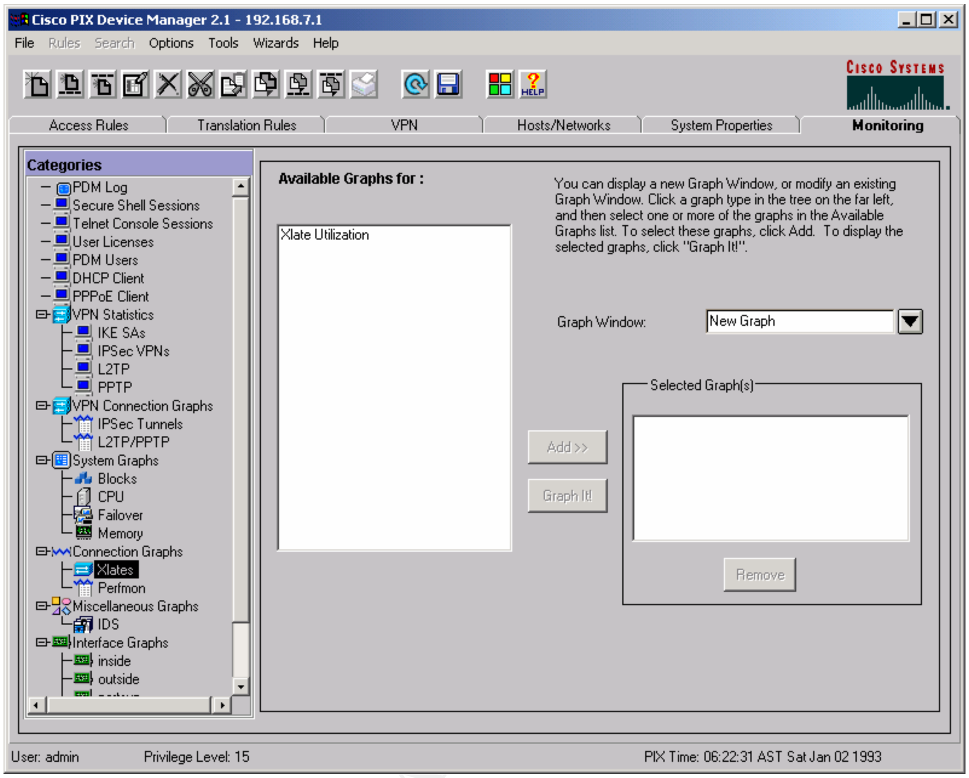
Interfaces

Select an interface from the table to edit its parameters.

Enabled	Hardware ID	Speed	Interface Name	Security Level	IP Address	Subnet Mask
Yes	ethernet1	100full	inside	100	192.168.7.1	255.255.255.0
Yes	ethernet5	100full	sync	90	192.168.2.1	255.255.255.0
Yes	ethernet3	100full	web	50	192.168.5.1	255.255.255.0
Yes	ethernet2	100full	partsup	50	192.168.4.1	255.255.255.0
Yes	ethernet4	100full	vpn	20	192.168.3.1	255.255.255.0
Yes	ethernet0	100full	outside	0	223.223.223.7	255.255.255.0

User: admin Privilege Level: 15 PIX Time: 06:21:41 AST Sat Jan 02 1993

Diagram 12 – PIX Interface Configuration



Xlate Utilization

You can display a new Graph Window, or modify an existing Graph Window. Click a graph type in the tree on the far left, and then select one or more of the graphs in the Available Graphs list. To select these graphs, click Add. To display the selected graphs, click "Graph It!".

Graph Window: New Graph

User: admin   Privilege Level: 15   PIX Time: 06:22:31 AST Sat Jan 02 1993

Diagram 13 – PDM Monitoring Screen