Global Information Assurance Certification Paper

Copyright SANS Institute
Author Retains Full Rights

This paper is taken from the GIAC directory of certified professionals. Reposting is not permited without express written permission.

Firewalls, Perimeter Protection and VPNs
GCFW Practical Assignment
Version 1.6a

By: Robert D. Nelson
Submitted on: February 22nd, 2002

© SANS Institute 2000 - 2005, Author retains full

© SANS Institute 2000 - 2005 Author retains full rights.

Preface:

To assist managers and other employees in reading this proposal, I have included two additional pages at the beginning of each section. The first page consists of the assignment and requirements of the section. The second page contains an outline of the following section as well as the specific requirements of the assignment that it satisfies, denoted as quoted sections in parentheses.

The outline only serves to emphasize where a reader may look for the satisfaction of a specific requirement. Other points may be covered in the same section or the same point may be covered in multiple sections.

© SANS Institute 2000 - 2005, Author retains full rights.

Assignment 1 – Security Architecture (15 points)

Define a security architecture for GIAC Enterprises, an e-business which deals in the online sale of fortune cookie sayings.

Your architecture must consider access requirements (and restrictions) for:

- *Customers (the companies that purchase bulk online fortunes)*
- *Suppliers (the authors of fortune cookie sayings that connect to supply fortunes)*
- *Partners (the international partners that translate and resell fortunes)*
- *GIAC Enterprises (the employees located on GIAC's internal network).*

You must explicitly define how the business operations of GIAC Enterprises will take place. How will each of the groups listed above connect to or communicate with GIAC Enterprises? How will GIAC employees access the outside world? What services, protocols, or applications will be used?

Defining what type of access is required and why is a critical part of this assignment. If you have not thought through how this access will take place, you will not be able to adequately define your security policy and ACLs/rulesets later in the paper.

In designing your architecture, you must include the following components:

- *filtering routers*
- *firewalls*
- *VPNs to business partners.*

Your architecture may also include the following optional components if they are appropriate to your design:

- *internal firewalls (are internal firewalls appropriate for additional, layered protection; to segment internal networks…?)*
- *secure remote access (is additional remote access required by administrators, salespeople, telecommuters…?)*

Include a diagram or set of diagrams that shows the layout of GIAC Enterprises' network and the location of each component listed above. Provide the specific brand and version of each perimeter defense component used in your design. Finally, include an explanation that describes the purpose of each component, the security function or role it carries out, and how the placement of each component on the network allows it to fulfill this role.

The network can be as complex or as simple as you like as long as it meets the functional requirements that you define according to the guidelines given above. The important thing is not how elaborate your network is, but that your design actually works.

© SANS Institute 2000 - 2005, Author retains full rights.

Section 1 Layout

1. Define the business requirements of GIAC Enterprises, including Customers, Suppliers, Partners, Employees, and Misc. (“explicitly define how the business operations of GIAC Enterprises will take place”)

2. Provide brief description and a complete diagram of the Network Architecture. (“Include a diagram or set of diagrams that shows the layout of GIAC Enterprises’ network and the location of each component listed”)

3. Discuss components of Network Architecture: (“Defining what type of access is required and why” and “include an explanation that describes the purpose of each component, the security function or role it carries out, and how the placement of each component on the network allows it to fulfill this role.))

    - Remote Access (“VPNs to business partners” and “secure remote access”)
    - Service Network
    - Firewall (“firewalls”)
    - Internal Network
    - Intrusion Detection
    - Internet Connection (“filtering routers”)
    - Infrastructure (“Provide the specific brand and version of each perimeter defense component”)
    - Exceptions

4. Summary of business plan and network architecture.

© SANS Institute 2000 - 2005, Author retains full rights.

Section 1: GIAC Enterprises Business Plan and Security Architecture

Business Plan:

GIAC Enterprises (Hereafter GIAC) is a fledgling company, located in State College, PA, that is in the business of selling fortune cookie sayings. GIAC does not provide the sayings, but handles direct and indirect sales to customers, as well as translation into foreign languages for overseas and cultural markets. The dot.com startup companies have provided a harsh lesson in how not to manage a company. While the company expects to increase in size dramatically over the next 12 to 18 months, fiscal matters are still being handled in a conservative fashion, with an eye for expandability rather than immediate expansion. The following pages outline a proposal for a flexible and expandable network that facilitates the day-to-day operations of GIAC, connecting GIAC to its suppliers and partners, on a conservative budget and with a mind for security.

An outline of the business process is as follows:

- Suppliers provide fortune cookie sayings to GIAC via secure web interfaces, CD-ROM (postal mail), and occasionally through email to an employee.
- GIAC then allows certain partners access to the sayings for translation. The fortunes are emailed to the partners, mailed on CD-ROM, or downloaded off a secure portion of the web site, depending on volume.
- Translated sayings are returned by the same methods, but not necessarily in the same method they were shipped out; i.e. A partner may pick up sayings online, but ship them back on CD, or vice versa.
- Customers and resellers will pick up their fortunes by either the web interface or a mailed CD-ROM. GIAC will not provide sayings in email format for customers or resellers.

Suppliers:

Suppliers need to be able to upload files, but will never be allowed to download them. Incoming files must be labeled with a date and timestamp, number of sayings, language(s), the company who uploaded them and (where applicable) the person who uploaded them. Along with uploading, the suppliers must be able to view previous transactions, including uploads, credits, and debits. The transaction log will show date, timestamp, number of sayings, language(s), the company, the user who uploaded the fortunes, and a transaction number. Credit and debit entries will show date, timestamp, US dollar amounts, company, and a brief explanation of the credit/debit. Credits will include the transaction numbers for the fortunes they are paying against. Debits will include a reason why, such as a correction of a previous credit.

These functions will be achieved by a unified web access system on a secure web server. The user will sign in with a company name, a user name (in small and one-person companies, this will be the same) and their passphrase. They will initially see a menu with the options “Upload fortunes” and “View transactions” which will perform the functions mentioned above.

To handle files that are transferred on CD-ROM or in email, GIAC employees will be able to use a similar web interface, described later, to upload the sayings as their employee ID while designating the correct user and company for the transaction log’s record.

Partners:

Partners need to access a wider variety of services. First, partners need to download files containing sayings. The files will have to be moved by GIAC employees from the supplier receive queue into each partner’s send queue. After downloading and translating, the partner needs to be able to upload the sayings. They will also need access to an area where an interested translator can claim unassigned groups of sayings. As with suppliers, the partner companies will have access to a transaction log. The transaction log will use the same format used for the suppliers, but with an additional field denoting whether a transaction was an upload or a download.

© SANS Institute 2000 - 2005 Author retains full rights.

Web access will be granted, similar to what suppliers receive, and transfers via email and CD-ROM will be handled in the same manner as GIAC employees handle it for suppliers.

Customers:

The customers interface will be most similar to that of the suppliers. Customers will begin by searching all sayings. Searches will accept the following parameters: languages, target audience, price per fortune, and number of fortunes. For example, a customer could start a search by specifying Chinese fortune cookies written in English, a price less than or equal to $.10 a fortune, and a total number of fortunes between 500 and 1000. Fortunes will be available as downloads or mailed on a CD-ROM for an additional fee. Finally, customers will be able to view their own transaction log that details purchases, method of delivery (download or CD-ROM), GIAC's inventory number for the group of sayings, and the cost. Debits and credits will be tracked as separate items.

Employees:

GIAC employees will have access to the fortune sayings database and files from either an internal database application or an internal website similar to what customers, suppliers, and partners see. The database application will be a native database application, such as an Oracle SQL app., designed to work like the web interface to keep a consistent look and feel. Employees will only be allowed to access their employee logins on the internal web server. To simplify security, the external web server will not allow employee logins and the internal web server will only allow employee logins. The two servers provide otherwise identical service. This separation is made to reduce the chances of stolen passwords causing damage to the database. Employees, if need be, can log in as customers with the database application. Additionally, employee logins will see an additional field when looking at the transaction log that specifies who made the update, an employer or the account owner.

Remote users, travelling salesmen and other employees requiring remote access will not be able to use the external web site or run the database application over the internet. To provide these users with access to their required tools without compromising the security of the network, they will connect to a Windows 2000 server running Citrix Metaframe. The Metaframe server will have all programs loaded that users might need when they are out of the office, such as the database app., a browser that can access the internal web server, office productivity suites (such as Corel WordPerfect or Microsoft Office), and other programs as needed. The server can be accessed over the internet, if the user has their own connection to the internet, or dialed into directly with a modem. This will allow users to run their programs remotely and still have access to files stored on the internal file server.

Employees will also be allowed to browse the web. Currently, GIAC operates on a "good faith" policy providing unrestricted access that relies on employees using their own best judgement. This design, however, includes room for the future use of monitoring software, in the event that the internet policy becomes stricter.

Other:

- Web access was chosen for customers, suppliers, and partners for a variety of reasons. By using a browser to access GIAC's site, users are given a familiar interface in which to do their work. Designed correctly, the site can also be viewed under any OS/browser combination, eliminating the need to restrict browsers that can view the site or to develop cross-platform applications. Users can also access GIAC's web page from any machine with a browser, simply by remembering *www.giac.com*. No software needs to be loaded on the target machine, easing desktop administration for our clients and partners. Traffic can also be encrypted via SSL.

- Metaframe is a very powerful solution for remote employee access. Like web access, it provides a consistent interface for the users. There are also a variety of clients for most popular operating systems and a less-powerful web browser client (ex: *http://metaframe.giac.com:81/applications* will show a listing of apps a client can run in their browser). Citrix now includes 128-bit RSA encryption for sessions, providing a secure encrypted communications channel. Metaframe runs on Windows 2000, which provides a familiar environment for administration.

© SANS Institute 2000 - 2005

Author retains full rights.

- Security is a vital concern for all companies as we enter 2002, and it is only becoming more important with time. Security is an integral feature and it must not be designed "after the fact" or made optional in any way. The foundation of GIAC's services must be built with security in mind. Not only are there legal repercussions both being a victim of a cyber-attack and by being used to attack another victim, but it has been shown time and again that a lack of security can severely damage a company's ability to make money. To this end, the above business plan was designed to enhance security. By narrowing external access to only three vital services – external web access for end users, Metaframe access for remote users, and web browsing for employees – these services can be focused on exclusively to avoid complicated rules detailing internal and external access. GIAC can also avoid the pitfall of having a single point of failure by building redundancy into the network at all points. Finally, the design builds in room for expansion. Whether it is another publicly available service, more servers, or even another switch, the network will be able to easily integrate the new features.

Network Design:

The design of a network is as important as what runs on it. A well-designed network makes users' lives easier, not more difficult. It helps protect the network from the users, not just the bad guys. Flexibility, redundancy, capacity and security are crucial aspects of the design.

A flexible network can save costs by a great deal. Flexibility ensures that adding a new infrastructure device to the network does not require replacing every other device at the same time. It also allows for a great deal of change in the topology of the network or in how applications work on the network without changing the infrastructure components.

Redundancy ensures that the failure of one or even two physically disparate devices does not cause the network to fail. The level of redundancy can vary from area to area but can always be increased upon demand in a flexible network.

Capacity must meet the needs of the applications and data that operate over the wire. The network should be designed to support the apps that users want rather than thin apps being chosen because they will not exceed the limited bandwidth of a less flexible network. There should also be room for some growth so that small expansion does not cause the network to become congested.

Thorough security greatly protects the investment in the network. It ensures that good-natured employees cannot "accidentally" find their way into a device and disrupt the network as much is it deters crackers from quick hit-and-run attacks on the network. In the case of network-based attacks, a well-placed monitoring station can also assist in tracking down the attacker for later prosecution or retrieval of stolen information.

Combined, the above attributes provide a powerful network that can be shaped by IT/MIS for their needs. The initial investment is well protected through flexibility and security, it allows users to choose the best applications for their jobs through capacity, and redundancy ensures little to no unscheduled downtime due to attacks or simple hardware failure.

The following diagram outlines the proposed network design for GIAC Enterprises. Each section will be described in its entirety as well as its purpose in the "big picture".

© SANS Institute 2000 - 2005 Author retains full rights.

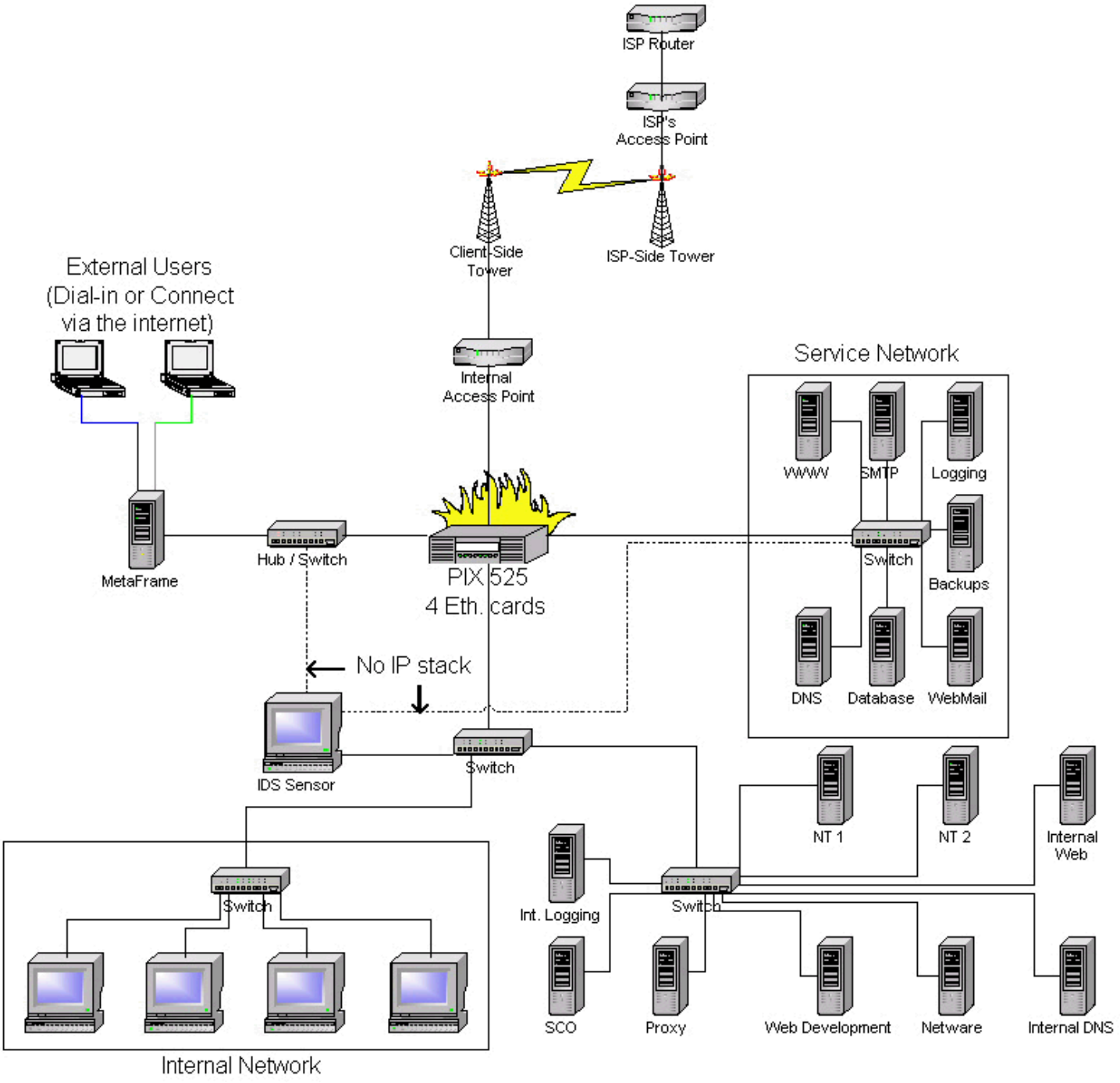


Figure 1.1 – GIAC Enterprise's Network Diagram

© SANS Institute 2000 - 2005

Author retains full rights.

Remote Access:

There are three types of connections for remote users, all of whom are employees. All remote users will log into a Citrix Metaframe server (center-left area of Fig. 1.1). The server runs Windows 2000 SP2 and Citrix Metaframe 1.8 SP3. The users can dial-in on a modem directly connected to the server (RAS), connect with a Citrix client over the internet, or use NFuse's web interface for published applications. The Metaframe server will exist in its own workgroup and users will only be created when requested and accompanied with a department head's authorization. Metaframe users will be able to use the same password as they use internally, to facilitate authentication between Metaframe and the internal systems, but will be subject to stronger password auditing rules. This includes, but is not limited to: 30 day expiration of passwords; requirements of 8 characters, minimum two capital letters, two digits, and one special character; and finally a temporary removal of remote privileges if the password is revealed by a password-cracking program within 30 days.

The Metaframe server will communicate with and run login scripts, where applicable, to the internal systems. This consists of logging into a Novell 5.1 server, access to a SCO Openserver 5.0.5+ database server, and access to various NT/2000 servers. Users will also be able to use Metaframe to browse the web as if they were sitting at their office desk, using whatever digital certificates or stored cookies they might have loaded. Microsoft Office 2000, Corel Office 2000, and other desktop applications will be loaded as required.

These features allow any user access to their own desktop and files from anywhere around the world, in a highly secure (128-bit RSA, ICA authorization) manner. Because users must have authorization to use Metaframe, it can be controlled exceptionally well. For administration, it is another NT server that can be maintained easily rather than an obscure proprietary black box that requires an expensive consultant to modify and examine.

Metaframe will back itself up using BackupExec 8.5 for NT. It has a 12/24 DDS3 drive. Since its primary function is to allow users to access files residing on machines other than itself, it will only run one backup per week. Each tape will be marked with a week and reused every year. This will provide 52 weeks of tapes to restore from, if need be.

Service Network:

The service network includes all of GIAC's publicly available services. These currently include, but are not limited to, webmail for travelling users, DNS, SMTP, our web access app, and a logging server for the machines on this network (Metaframe will also use this as its logging host).

DNS and SMTP, services for name resolution and mail service, are two publicly accessible and *very* important services without which the company could not function. DNS allows people easy access to GIAC's services (www.giac.com, webmail.giac.com, metaframe.giac.com, etc) instead of having to remember difficult sequences of numbers (1.89.72.34, 1.92.14.5, etc.). DNS attacks and simple failures have caused extended "outages" of sites in the past where the site was unaffected but a DNS foul-up prevented anyone from getting to the site (Microsoft: http://www.microsoft.com/presspass/press/2001/Jan01/01-24DNSpr.asp). The SMTP server allows GIAC to send and receive mail. If it is not running or is down for an extended period of time, it will not allow people to send mail to, or receive replies from, anyone at GIAC. Together, these are two services that greatly affect GIAC's ability to do business on the internet and need to be guarded as best as possible. This defense is detailed later, in section 2.

The DNS and SMTP servers both run on different Red Hat 7.2 servers, one running BIND 9.1.3-4 and the other running Sendmail 8.11.6-3, with regular updates, of course. The mailserver has unlimited access for SMTP only out to the internet and will be used by GroupWise, on the Novell server, for outgoing services. The DNS server's access is restricted to allow full access to domain/udp but only allowing domain/tcp between it and the secondary DNS server. The Metaframe server also uses this DNS server; an allowance for domain/udp between the two machines is required. The secondary DNS server for giac.com will be provided by Get Wireless.

Webmail, running Captaris Webmail v3.62 (www.captaris.com) on Windows 2000, may seem like a superfluous service because of Metaframe's ability to provide access to email. However, webmail serves two additional functions. Unlike Metaframe, GIAC does not restrict employee's access to webmail – everyone who has a mail account also has webmail access. It also serves as a redundant backup system in case Metaframe fails or is down for maintenance. Users can

© SANS Institute 2000 - 2005 Author retains full rights.

often be without access to Microsoft Word or their files for a few hours or even a day, but losing access to e-mail for more than a few hours can significantly affect their ability to perform their daily tasks. Any salesman who loses access to email may not be able to respond to a customer's request for a large order, or perhaps a translator on another continent may use email to communicate with an employee because of a large time difference. While not as vitally important as DNS or SMTP, webmail is a backup system and should be kept secure and accessible 24/7.

Of equal importance to DNS, web access is GIAC's primary application on the service network. It is only through this web access that GIAC can continue to make money. If partners, suppliers, and customers are denied access for extended periods of scheduled maintenance or unscheduled downtime, income will be affected. This service, along with the DNS server that allows people to connect to it, is the lifeblood of GIAC. No expense should be spared in the protection of web access from crackers, internal attacks from disgruntled employees, virus attacks, or accidental attacks through incorrect configuration or updates. A very strict set of firewall rules is implemented to protect access to the server itself. A policy entitled "Client Web Application Server Policy" details the rules for access of this server by employees – what departments have access to it, how changes are tracked, how versioning is implemented, and any other rules that may need to be implemented. Filesystem rights are strict, allowing only certain users access to the directories containing the web application. Auditing is enabled whenever possible to record all events, regardless of whether it's an attack or an incorrect update.

The web access server will only run a web server; the actual database will be on a separate machine. This helps prevent simple attacks that can gain access to the webserver from automatically gaining access to the database as well. Segregation of the database and the web server also allows GIAC to set up multiple web servers not only in a server farm, but with a different internal web server running a different version of the web access software. The web server will be running Windows 2000 with IIS 5.0. The database will be hosted on Red Hat 7.2 running Postgresql 7.1.3-2.

The machines on the service network, plus the Metaframe server, will all use a central log server, also located on the service network. This will collect data such as the auditing on the web server, successful and failed authentication attempts on the webmail and Metaframe servers, changes to DNS, and other miscellaneous tracking information the servers collect. The servers (including Metaframe) will use the same publicly available stratum 2 NTP server, located geographically close, for time synchronization.

All the machines will be backed up onto one of two machines. For Windows-based systems (web servers), a server running Windows 2000 and Backup Exec 8.5 for NT will provide nightly backups. Unix-based systems (database, DNS, SMTP, logging) will use Microlite's BackupEdge 01.02.00, build 4. BackupEdge and Backup Exec both offer remote backups to a variety of platforms – that is, a variety of Unix versions or Windows variants, respectively – with central reporting. Each machine will have a DLT library attached to it with sufficient room for the data on the systems plus 20% expected data growth. Currently, the Windows backup uses a 5-cartridge loader providing a total of 175/350GB capacity – far outstripping the 20% growth requirement, yet one of the smallest library units available. Unix backups will use a 15-cartridge loader with dual drives, providing approximately 525/1050GB, which should be sufficient to store the current database and logging for all GIAC's systems.

Backups will be rotated with two sets of weekly backups and 12 monthly backups. This will provide GIAC with a year's worth of data. Anything older than a year is not likely to be very useful, but GIAC may store random backup sets in a safety deposit box, "just in case." No tape will be backed up to more than 50 times before it will be replaced. Backups will be tested periodically – at least twice every six months, for every system being backed up. By taking older tapes out of the rotation, GIAC lowers the chance of a known good backup being fouled by a bad tape.

The Firewall:

The service network, the Metaframe server, and the internal network (described below) will all be protected by a PIX 525 running the latest version of PIX software (v6.1 currently). PIX hardware is reliable and sturdy, handles IPSec well, provides thorough logging information, is well documented by Cisco as well as third parties, and is quite affordable for the protection it offers. The PIX 525 was chosen because it can handle 4 ethernet connections (internal, external, Metaframe, and the service network) as well as an approx. traffic of 370Mbps, far above the daily traffic GIAC can expect to see for the next two to three years. At its relatively low price (approximately $50,000), a second PIX can be

© SANS Institute 2000 - 2005

Author retains full rights.

purchased for either failover redundancy – if one PIX becomes too busy or stops responding due to hardware or software issues, the other will pick up the workload – or as a spare in case of severe hardware failure. The price of a second or even third PIX that sits in storage in case of hardware failure is inestimably smaller than the price of completely losing internet access for any length of time. It is also very easy to find administrator's trained on Cisco equipment and to find books on the subject. A variety of access-lists, detailed in Section 2, provide quite a bit of flexibility and adaptability to deal with a wide range of security requirements.

The firewall's main purpose is to permit or allow connections. It will achieve this with a default policy of denying all connections and only permitting specified connections. This means a conscious effort must be applied to allow a connection that could be harmful, rather than allowing everything in and having to expend effort at locking down every service and connection that is not needed. The tangible benefits appear in the form of less effort expended in dealing with new cyber-threats, most of which are targeted at unused services that an "allow all" policy enables by default.

The secondary purpose of the firewall is to segregate data and networks. Rather than allowing all of GIAC's equipment to be treated as trusted systems, it will again deny all connections except those specifically allowed. The treatment of all GIAC-owned hardware as trusted systems allows an attacker who gains access to one machine to gain full access to the network. By segregating portions of the network, an attack on one machine can only gain access to that segment of the network without having to start new attacks on the other segments. For example, the internal systems will not be allowed to talk to machines on the service network, and machines on the service network will not be allowed to talk to the Metaframe server, but the Metaframe server will have access to the internal file server. The specific rules and policies will be discussed in Section 2.

Desktop users in GIAC offices will browse the web using a proxy. The firewall will provide Network Access Translation (NAT) service to protect the proxy from reverse attacks upon it, as well as protecting any desktops that have direct access to the internet, such as the security administrator's desktop. Potentially, if the open policy on internet access were ever to change, the PIX works well with applications like WebSense (http://www.websense.com/) for content-based analysis and logging to monitor internet usage. The PIX 525 has a fast CPU to handle content-based filtering through its own internal capabilities or through third-party add-ons, such as WebSense, for a large number of users – up to 280,000 simultaneous connections, according to Cisco (http://www.cisco.com/warp/public/cc/pd/fw/sqfw500/). While the existing internet policy is very open, the options for such future capabilities are already built in to the PIX. Other internet access, such as anti-virus updates, can be handled by having the programs use the proxy server without any additional firewall configuration. If future applications require direct internet access from desktops and will not work through a proxy server, additional allowances can be made on the firewall without requiring changes to how the desktops access the internet.

Finally, the PIX will do the lion's share of the logging and accounting functions. Every packet that comes into or leaves any of GIAC's properties will be processed by the PIX in some manner. Any attacks that make it past the firewall rules, internal users who make illicit access across the PIX, or even simple incorrect configurations on a server that flood the wires will be logged by the PIX. It will report everything to the logging servers on both the service network and the internal network. While not necessarily the most economical use of wire speeds, it provides the first level of redundancy of our logging, auditing, and monitoring systems. If bandwidth becomes strained, the PIX can stop logging to one or the other servers until available bandwidth is increased.

Internal Network:

The largest number of network-aware devices will reside on the internal network, but the lowest number of required internet services are on the inside. Therefore, the policy is to deny access from the other networks (Metaframe, service, and internet) to the internal network, specifying exceptions, and to provide most access from the inside to the internet. To restrict the ability of end users to flood GIAC's internet connection with decidedly non-business traffic (e.g. Napster, Morpheus, and hundreds of other peer-to-peer services, which also tend to involve a possibility of copyright violations), desktop computers will be set up to use a proxy server. The proxy server is a Red Hat Linux 7.2 server running Squid 2.4.STABLE1-5. No access lists are enabled on the proxy server; the firewall handles all access control. Any additional services that are not handled natively by Squid – http, secure, socks, and ftp protocols – will be handled by helper applications relevant to the protocol, such as an H.323 proxy application, as required for business needs.

© SANS Institute 2000 - 2005

Author retains full rights.

Internal users will not need access to any of the machines on the service network or the Metaframe server. Desktops will connect to a NetWare 5.1 server for file, print, and time synchronization (NTP) services and a Red Hat 7.2 server running a caching nameserver, caching-nameserver-7.2-1.noarch.rpm, for DNS service. The NetWare 5.1 server will only be allowed to talk to one machine on the internet for NTP synchronization, clock.psu.edu. Penn State requires an email to be sent requesting permission to use their servers. The BIND server will have restricted access for DNS; domain/udp for all hosts. These two servers will log to an internal logging machine – the number of attacks against these internal machines is hoped to be low, but ignoring log files until after a successful attack only helps the intruders. Any direct access to the internet will have to be approved by an employee's department head and by the firewall maintainer (currently the MIS department head). As per internal policies, such changes will be documented in full.

Finally, GIAC employees will have to access an internal web server running a special version of web access software. This version will only allow GIAC employee logins, as the outside version only allows non-GIAC logins. This web server will be running Windows 2000 as well as the web access software.

Intrusion Detection:

Security is a process, not a product. A system designed for security must be constantly analyzed and updated. Part of the analysis is intrusion detection. Firewalls are not foolproof, particularly when an attack is carried out on a channel that is vital to a business, such as an attack on a webserver so that it looks like one out of a million clients visiting the site. Intrusion detection analyzes packets that get past the firewall and attempts to reconcile them with valid traffic. Current intrusion detection technology is reactive; that is, it must look for known patterns of attack and will not detect new and undocumented attacks. If the intrusion detection system (IDS) is not constantly updated, it will only detect older attacks and none of the newer and more dangerous attacks upon a network. On top of this, IDS captures *all* packets that match an attack pattern, but it cannot directly determine if it is an actual attack or a false alarm. Much human intervention is required to search through IDS logs to see what dangers a network is in from what kind of attacks and, in the case of a successful attack, who the perpetrator was, how they were able to get in, and what damage was done.

Because of the high cost of human intervention in the processing of logs and proper maintenance of an IDS system, placement of IDS sensors is very tricky. On a high traffic line, the ratio of human time to log files can be as high as 4 hours for every 1-hour's worth of log files. Most companies can not afford 4 such employees to be able to sift through the logs in real-time, and GIAC is no exception. The IDS sensors are placed so as to catch the maximum amount of important data with the lowest number of false alarms. The main IDS system is set up to log to both itself and the internal logging server mentioned above, storing data from each sensor in different files for easier processing.

The IDS system is a machine running Red Hat 7.2 with Snort 1.8.3-5 as well as having four network cards. As per Figure 1.1, the machine is plugged into a hub on the Metaframe network, the switch on the service network, and into the backbone switch on the internal network. The solid line coming out of the IDS system represents the only network card that receives an IP address. The other two dashed connections have TCP/IP enabled, but no IP address. This is done to set the connections to a listen only mode, which eliminates the ability of anyone to attack the IDS system from the Metaframe or service network because the machine cannot respond. However, the IDS system has to have one connection with an IP in order to be able to communicate with the logging server as well as to receive updates to Snort and its rulesets (available at http://www.snort.org/downloads/snortrules.tar.gz). The fourth network card is a spare. It can be used in an emergency to replace one of the other three or, if a large number of attacks are detected, can be set up to monitor the area between the PIX and the internet router. This is a very high traffic area that will pick up a lot of false alarms, mostly because it is listening to traffic *before* the firewall has a chance to filter out any packets.

The IDS system and the logging server both have large hard drives to store the data on, currently over 100GB on each. The IDS logs will be spot-checked throughout the week and then backed up onto tape. The logs will then be wiped and collected for another week before backing them up on tape again. A full 52 sets of tapes will be used to store a whole year's worth of logs for future analysis, if need be.

Internet Connection:

Get Wireless is a wireless internet service provider located in State College, PA, not far from GIAC headquarters. For a

© SANS Institute 2000 - 2005

Author retains full rights.

reasonable price per month, plus less hassle than a phone company, they provide a stable connection from which to grow. While 2-11Mbps variable rate service is not likely to be adequate for more than 12 months at GIAC's current growth rate, Get Wireless provides a number of features that allows GIAC to have a solid internet presence until GIAC secures a faster connection. While wireless connections offer their own unique challenges, there are significantly less disadvantages than a landline offers – no-one is able to cut the line with a backhoe nor have a line short because of a rising water table (This was what Bell Atlantic *claimed* with GIAC's partial T1 last year!). Only the most severe of weather is likely to affect the wireless link, although it is a possible risk. For this reason, a DSL line has been secured from CEI Networks, another local provider. It is a significantly smaller pipeline than Get Wireless offers, but should suffice for a backup solution. It is worth noting that neither connection is provided by a large telecommunications company, most of which are known for unreliable service and horrible customer support.

Get Wireless also provides a secondary DNS service and two routers, one on either end of their wireless link. They have proven very cooperative in setting up some simple filters on their router that help with site security while lessening the strain on our PIX, such as simple ingress/egress filtering as well as block known unused networks (1.x.x.x, 5.x.x.x, etc.). This will be detailed in Section 2.

The PIX is set up to take advantage of both the primary wireless link and the secondary DSL link. The changes required are fairly simple, namely swapping the incoming wire on the PIX from wireless to DSL and changing the outside namespace. Two versions of the PIX setup will be maintained, one with the wireless namespace and one with the DSL namespace (always one version for the primary internet connection and another version for the secondary, regardless of the type of connection) that can be uploaded from an internal TFTP server. TFTP is slightly more reliable than having an administrator edit the PIX configuration manually because the TFTP file is not as likely to contain typos that will cause errors and because the process is fairly automated, allowing even a layman to make the change in an emergency. Slightly more overhead is incurred in keeping two files up-to-date with similar configurations, but the potential for loss vastly outweighs it.

GIAC's DNS zone (giac.com) will contain only addresses for servers that provide an external service. E.g. the DNS server may be known as dns.giac.com, but the logging server will have no name. Likewise, reverse-lookups will only be provided for those services that need it. For example, by not providing a reverse-lookup for a web server that is keyed to only respond to the DNS name, simple attacks on the IP address can avoid revealing the presence of a web server to would-be attackers. The secondary DNS server, operated by Get Wireless, must be secured as well. GIAC's DNS server disallows zone transfers to all but the secondary DNS server, preventing attackers from gaining an easy-to-read map of GIAC's network. If Get Wireless's server is not similarly secured, then GIAC's security is all for naught, as Get Wireless would provide the attacker with the map anyway.

As can be seen, it is important to find a service provider that will work with GIAC to help ensure the security of GIAC's systems and data. An irresponsible provider can quickly undo the most elaborate prevention mechanisms by providing would-be attackers with the data that GIAC is protecting. Currently, Get Wireless is willing and able to work with GIAC; CEI Networks, another local provider, is able but not quite so willing. Future providers *must* be at least as cooperative as Get Wireless. Because of their important role in securing GIAC's assets, the service provider must also be part of all security analysis and audits if the audits are to be of any worth to GIAC.

Infrastructure:

Many companies choose to standardize their equipment, frequently on Cisco. While there are many benefits to standardization, there are also quite a few disadvantages. If a company standardizes on Cisco, they are assured that there is a large pool of Cisco-qualified experts from which to draw talent. However, it is quite easy for even the largest company to find themselves locked into a vendor's products with no easy way out and with costs rapidly rising. A homogeneous network also finds the infrastructure vulnerable to a single attack, where a heterogeneous network may only lose a small section to one attack. By standardizing on equipment where it makes sense, such as all routers of a certain type, one can gain many of the advantages without many of the disadvantages, most notably vendor lock-in.

GIAC will standardize on the type of firewalls and switches in use as well as the server vendor. All firewalls will run the latest Cisco PIX software, now at version 6.1. The switches will be from HP's ProCurve lineup. ProCurves are switches

© SANS Institute 2000 - 2005

Author retains full rights.

that provide the same abilities as comparable Cisco switches but at a fraction of the cost. They are layer three switches, providing VLAN support, trunk lines, and web management. HP also provides a lifetime warranty on all ProCurve products along with frequent rebates. The ProCurve 2524M is the 24 port version of the switch and the 4000M is a 10 slot chassis filled with 5 8-port cards, providing 40 ports out of the box; both run at 10/100Mbps speeds. The 8000M is an empty 10 slot chassis that filled with 10/100 8-port cards, fiber modules, gigabit ethernet ports, and a variety of other modules that are of use to most companies. Along with higher-end rack-chassis systems in the ProCurve lineup, they provide very functional and solid building blocks for the backbone of the network. Finally, all servers will be Dell PowerEdge systems. Dell offers a variety of rackmount systems, helping to save floorspace and electricity, and they offer flexible rack-mounted solutions, including a console switch (180ES or 2160ES) and a flatscreen with attached keyboard/mouse (Dell 1U Flat Panel Monitor) to help save even more floorspace. The PowerEdge server lineup scales very well from small departmental servers (PE 500 and 1x00-series) up to very high-end systems with high-availability and a full complement of powerful components for a company-available system (PE 8xxx-series).

Exceptions:

To remain a flexible network, it must be possible to use products and setups that may violate the above policies. To follow the policies as if they are set in stone would only be a hindrance to the users, whose needs will change over time in entirely unanticipated manners. The policies that protect the network from incidental or direct damage are useless if they prevent GIAC employees from being able to work.

For example, a stated policy is that no machines on the service network will have access to the internal network. While this is a good security policy, it does not provide an easy way to allow web developers to update GIAC's web site or change web access policies for suppliers. Other similar exceptions would be mail and DNS – if the systems administrators cannot access these machines, then they will be prevented from managing mail accounts and the DNS information. The database server needs to talk with both the external web access server and the internal web access server. Last, but not least, analyzing the logs of the service network will be much easier if the administrators can access the logging server from their machines.

Access to these machines will be provided by SSH, for shell access, or SMB protocol, for network shares. To access the PostgreSQL database, the firewall will permit PostgreSQL connections from the internal web server to the service database network. NDS and NCP protocols will be permitted from the Metaframe server to the internal NetWare server, and it will also allow Metaframe web access to the internal web server.

There are other exceptions to be expected. The systems administrators, for instance, will require direct access to the internet, rather than through the proxy server, to test new programs/protocols or to allow them to do work that may require taking the proxy offline. A central machine may be used to download and store anti-virus updates that can't be handled with the proxy. End-users, in general, may find a need in the future for a program that doesn't work through proxy, such as certificates or some sort of business-related peer-to-peer program, but requires direct internet access.

There are plenty of reasons why the security policies should be well defined but also very flexible. Six months is a seeming eternity in the high-tech industry and forcing the same policies upon users year after year is only likely to stifle new technologies that may improve the business process.

Summary:

By carefully spelling out the complete requirements of GIAC Enterprises – from the needs of the employees to the requirements of working with suppliers and to handling individual sales of the finished product – a detailed network architecture can be developed. Defining policy based on these needs and open to future needs ensures that the architecture is utilized and the investment is protected for its lifetime. With a solid architecture and a generic policy defined, a specific implementation plan can be spelled out in the next section.

© SANS Institute 2000 - 2005 Author retains full rights.

Assignment 2 – Security Policy (35 points) Based on the security architecture that you defined in Assignment 1, provide a security policy for AT LEAST the following three components:

- *Border Router*
- *Primary Firewall*
- *VPN*

You may also wish to include one or more internal firewalls used to implement defense in depth or to separate business functions.

By "security policy" we mean the specific Access Control List (ACL), firewall ruleset, IPSec policy, etc. (as appropriate) for the specific component used in your architecture. For each component, be sure to consider the access requirements for internal users, customers, suppliers, and partners that you defined in Assignment 1. The policies you define should accurately reflect those business needs as well as appropriate security considerations.

You must include the complete policy (explicit ACLs, ruleset, IPSec policy) in your paper. It is not enough to simply state "I would include ingress and egress filtering…" etc. The policies may be included in an Appendix if doing so will help the "flow" of the paper.

(Special note on VPNs: since IPSec VPNs are still a bit flaky when it comes to implementation, that component will be graded more loosely than the border router and primary firewall. However, be sure to define whether split-horizon is implemented, key exchange parameters, the choice of AH or ESP and why. PPP-based VPNs are also fully acceptable as long as they are well defined.)

In addition, for one of the three security policies defined above, you must incorporate a tutorial on how to implement the policy. Use screen shots, network traffic traces, firewall log information, and/or URLs to find further information to clarify your instructions. Be certain to include the following:

1. *A general explanation of the syntax or format of the ACL, filter, or rule for your device.*
2. *A general description of each of the parts of the ACL, filter, or rule.*
3. *An general explanation of how to apply a given ACL, filter, or rule.*
4. *For each ACL, filter, or rule in your security policy, describe:*
    - *the service or protocol addressed by the rule, and the reason this service might be considered a vulnerability.*
    - *Any relevant information about the behavior of the service or protocol on the network.*
    - *If the order of the rules is important, include an explanation of why certain rules must come before (or after) other rules.*
5. *Select three sample rules from your policy and explain how you would test each rule to make sure it has been applied and is working properly.*

Be certain to point out any tips, tricks, or potential problems ("gotchas").

© SANS Institute 2000 - 2005, Author retains full rights.

Section 2 Layout

1. Describe the components and information such as IP addresses used for the border router, primary firewall, and the VPN ("provide a security policy for AT LEAST the following three components…").

2. Border Router

   - Policy ("By security policy we mean the specific Access Control List (ACL), firewall ruleset, IPSec policy, etc. for the specific component used")
   - Tutorial applying policy and explaining service functions ("In addition, for one of the three security policies defined above, you must incorporate a tutorial on how to implement the policy" and "For each ACL, filter, or rule in your security policy, describe: the service or protocol addressed by the rule, and the reason the service might be considered a vulnerability; any relevant information about the behavior of the service or protocol on the network; if the order of the rules is important, include an explanation of why certain rules must come before (or after) other rules.")

3. Primary Firewall

   - Policy ("By security policy we mean the specific Access Control List (ACL), firewall ruleset, IPSec policy, etc. for the specific component used")
   - Tutorial applying policy and explaining service functions ("In addition, for one of the three security policies defined above, you must incorporate a tutorial on how to implement the policy" and "For each ACL, filter, or rule in your security policy, describe: the service or protocol addressed by the rule, and the reason the service might be considered a vulnerability; any relevant information about the behavior of the service or protocol on the network; if the order of the rules is important, include an explanation of why certain rules must come before (or after) other rules.")

4. VPN

   - Policy, including filesystem and user creation ("By security policy we mean the specific Access Control List (ACL), firewall ruleset, IPSec policy, etc. for the specific component used"). No tutorial provided.

5. Notes on the above policies, including citations and exceptions. ("Be certain to point out any tips, tricks, or potential problems")

6. Testing – how to show that the rules are applied by way of testing three examples ("Select three sample rules from your policy and explain how to test each to make sure it is applied")

7. Summary of device policies

© SANS Institute 2000 - 2005 Author retains full rights.

Section 2: Security Policy In-Depth

The security of three distinct components of GIAC's network architecture will be described in the following section. The first is slightly unique in that the internet provider, Get Wireless, owns and maintains GIAC's border router. GIAC does have a limited staff, and removing the border router from GIAC's jurisdiction allows staff to focus on more variable pieces of the network that require their attention. It also saves some money in equipment and support – Get Wireless has specified in their provider contract that they will handle all hardware costs, upgrades, and support for the border router, a pricetag that could easily rise very quickly.

The second component analyzed is the PIX 525, GIAC's firewall. It permits and allows access between the internet, Metaframe, the service network, and GIAC's internal network based on the policies described in Section 1. The PIX 525 will be entirely under GIAC's jurisdiction to guarantee the integrity of its setup and maintain full control over the traffic it manages.

Metaframe's policies will also be explained. Unlike the other two devices, Metaframe's policies are very small. Access to and from the machine will be protected *by* the firewall and router, so most of its policies will deal with ensuring that users accessing it can not damage the underlying OS or other user's files and setup.

To protect the security of the architecture the real IP scheme being used will not show up in this document. GIAC's public space will be represented by 200.200.200.0 / 255.255.255.128 and the internal networks will be represented by 10.200.0.0 / 255.255.0.0 (internal), 10.201.0.0 / 255.255.255.128 (metaframe), and 10.202.0.0 / 255.255.255.0 (service). Get Wireless's IP block is 199.199.0.0 / 255.255.0.0, divided by their own requirements.

Border Router (owned by Get Wireless):

This router is a Cisco 1605R router running IOS 12.2(6)a, kept updated with the latest version of IOS. As stated in Section 1, the policy on the router is to perform checks against IP spoofing and blocking IP "black hole" networks – networks that have are provided through Ipv4 but are currently unassigned to any company. This will protect GIAC from untraceable packets, whether they are malicious attackers or benign users set up incorrectly, and it will leave it up to the PIX to filter out traffic for specific attacks or unused services. Having each device perform only one type of analysis helps keep the load of each device as low as possible while still protecting the network.

The router will have an external IP of 199.199.199.126 / 255.255.255.64. The internal interface between it and the PIX will have the IP 200.200.200.1 / 255.255.255.128

The initial config of the router includes the following commands, which change default values:

```
service password-encryption
enable secret <password>
logging buffered
logging 200.200.200.25
no ip source-route
no service finger
no cdp run
no service tcp-small-servers
no service udp-small-servers
no ip http server
no ip bootp server
access-list 1 permit host 200.200.200.3
line vty 0 4
password <password>
access-class 1 in
banner /
```

© SANS Institute 2000 - 2005, Author retains full rights.

WARNING: Authorized access for GIAC system administrators only!
/

Now, specify the access-lists, 20 for incoming and 50 for outgoing

```
access-list 20 deny 10.0.0.0 0.255.255.255 log
access-list 20 deny 172.16.0.0 0.15.255.255 log
access-list 20 deny 192.168.0.0 0.0.255.255 log
access-list 20 deny 127.0.0.0 0.255.255.255 log
access-list 20 deny 224.0.0.0 31.255.255.255 log
access-list 20 deny 0.0.0.0 0.255.255.255 log
access-list 20 deny 1.0.0.0 0.255.255.255 log
access-list 20 deny 2.0.0.0 0.255.255.255 log
access-list 20 deny 5.0.0.0 0.255.255.255 log
...
access-list 20 deny 219.0.0.0 0.255.255.255 log
access-list 20 deny 220.0.0.0 3.255.255.255 log
access-list 20 deny 200.200.200.0 0.0.0.128 log
access-list 20 permit any
access-list 50 permit 200.200.200.0 0.0.0.128
access-list 50 deny any any log
```

The complete set of commands run to secure Get Wireless's router for GIAC will include some per-interface commands, attaching access-lists to their interfaces as well as some other per-interface settings:

```
int ethernet1
ip address 199.199.199.126 255.255.255.64
no ip directed-broadcast
ntp disable
no cdp enable
no ip unreachables
no ip redirects
ip access-group 20 in
no snmp
int ethernet0
ip address 200.200.200.1 255.255.255.0
no ip directed-broadcast
no cdp enable
no ip unreachables
no ip redirects
ip access-group 50 in
no snmp
write memory
```

Running the command "show config" will display the settings that are in place. A copy of what is produced by this command is in the appendix under the entry "Router Config".

Explanation of router config and tutorial:

The router configuration is fairly simple, being that it is acting as a traffic manager, not giving more than a cursory glance at the traffic as it passes or is blocked. The tutorial will display a prompt and commands to be entered from it followed by an explanation of the commands. The prompt will only be displayed when it changes.

The first step is to log in to the router and enter "enable" to gain access to the enable-level prompt. There is no initial

© SANS Institute 2000 - 2005 Author retains full rights.

© SANS Institute 2000 - 2005 Author retains full rights.

password.

```
gw-giac>enable
Password:
gw-giac#
```

Now that the user has entered enable level, queries can be made and config mode can be entered. The commands will be entered in the following sequence, starting with entering configuration mode:

```
gw-giac#config term
gw-giac(config)#service password-encryption
        enable secret <password>
        logging buffered
        logging 200.200.200.25
        no ip source-route
        no service finger
        no cdp run
        no service tcp-small-servers
        no service udp-small-servers
        no ip http server
        no ip bootp server
        access-list 1 permit host 200.200.200.3
        line vty 0 4
        password <password>
        access-class 1 in
        banner /
        WARNING: Authorized access for GIAC system administrators only!
        /
```

The first commands set encryption on the password, then specify the password (not shown here). Logging is set to buffered to prevent high throughput traffic from wiping entries out of the log before they can be sent to the remote logging server, specified in line 4 as 200.200.200.25. Then, ip source routing is disabled. Source routing can allow an attacker to force traffic to return down a different path than it was sent, possibly passing it by a host that the attacker may have compromised and can use to hijack a session or sniff the contents. The service “finger” is disabled, as well as the Cisco Discovery Protocol (CDP), “small tcp/udp servers” under port 20 that are typically used for attacks, a web interface to the router, and the bootp service. None of these services are required for GIAC’s business on the internet and are only useful in assisting an attacker in mapping GIAC’s network for further attacks. The next four lines – access-list, line, password, and access-class – restrict access to the router’s console to the IP address 200.200.200.3 with the specified password. GIAC systems administration machines will connect with this IP address. Get Wireless will only attach to the firewall at the physical console port.

The last line, banner, is a warning statement. If someone were to connect to the router and attack it, this is a legal statement disallowing an attacker (or even an employee!) from unauthorized access. It has yet to be tested in court, but some lawyers theorize that absence of a disclaimer statement may be a technicality that would allow an attacker to walk. It does not slow the machine down and removes a possible legal block, so the banner will always be enabled.

Access lists can be of three types: standard acl, extended acl, or reflexive acl. Standard acl’s, which are access-lists with a number of 1-99, check packets and deny or permit based on the source IP only. Extended ACL’s and reflexive ACL’s check on the source and target IP or the presence of a state table entry for a connection initiation, respectively. Standard ACL’s are much better suited to this router because of how they only check on source IP – the only requirement in GIAC’s stated policy. The following ACL’s are entered:

```
gw-giac(config)#access-list 20 deny 10.0.0.0 0.255.255.255 log
        access-list 20 deny 172.16.0.0 0.15.255.255 log
```

© SANS Institute 2000 - 2005, Author retains full rights.

```
access-list 20 deny 192.168.0.0 0.0.255.255 log
access-list 20 deny 127.0.0.0 0.255.255.255 log
access-list 20 deny 224.0.0.0 31.255.255.255 log
access-list 20 deny 0.0.0.0 255.255.255.255 log
access-list 20 deny 1.0.0.0 255.255.255.255 log
access-list 20 deny 2.0.0.0 255.255.255.255 log
access-list 20 deny 5.0.0.0 255.255.255.255 log
...
access-list 20 deny 219.0.0.0 255.255.255.255 log
access-list 20 deny 220.0.0.0 3.255.255.255 log
access-list 20 deny 200.200.200.0 0.0.0.128 log
access-list 20 permit any
```

Access list 20 is for inbound traffic. The first five lines deny access to any packets with a source IP of a non-routable network (RFC 1918, Address Allocation for Private Internets, http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc1918.html), the loopback network, and the multicast network. Non-routable network addresses should, by definition, not be routed across the internet to GIAC's doorstep, so packets from them are of dubious nature without even looking at their content or destination. Likewise, the loopback network should only be used for a machine to talk to itself, not across the internet. Multicast packets are not necessarily malicious packets, but often contain streaming media. Since no GIAC employee has a need to see all streaming media across the internet, it will be blocked. If there is a business reason to have access to this, employees may request this policy to be changed.

The next six lines are blocking packets from the aforementioned "black-hole" networks. A complete list will be provided in the appendices. The next to last line blocks traffic that is coming across the internet, with GIAC's addresses as a source, from getting past the router. It makes little sense for a GIAC machine to try talking to another GIAC machine by going out across the internet and back again. These packets represent either a grossly incorrect configuration of a GIAC machine or a malicious attack on GIAC and will be blocked.

All the deny statements are followed with the word "log", indicating that any packet that is matched by the statements – that is, any packet that is denied by the router – is logged. The last statement, which permits any and all other traffic, will not get logged. Logging every single packet coming into GIAC would simply overload the logging server with sheer numbers.

The next ACL consists of two lines:

```
gw-giac(config)#access-list 50 permit 200.200.200.0 0.0.0.128
        access-list 50 deny any any log
```

Access list 50 is designated for outbound traffic. All outbound traffic should have a source address that belongs to GIAC, so the first line permits such traffic. Any other traffic is malicious or from an incorrectly configured machine and will be stopped and logged.

```
gw-giac(config)#int ethernet1
gw-giac(config-if)#ip address 199.199.199.126 255.255.255.64
        no ip directed-broadcast
        ntp disable
        no cdp enable
        no ip unreachables
        no ip redirects
        ip access-group 20 in
        no snmp
gw-giac(config)#
```

Interface ethernet 1 is hooked up to the wireless receiver and is the router's connection to the internet. The first line

SANS Institute 2000 - 2005 Author retains full rights.

brings the router out of global mode into interface mode for ethernet 1. Line 2 gives ethernet 1 an IP address and activates the interface. Directed broadcasts are disabled in line 3; they are often used to perform "smurf" attacks ("The Latest in Denial of Service Attacks: "SMURFING" Description and Information To Minimize Effects" by Craig A. Huegen, http://www.pentics.net/denial-of-service/white-papers/smurf.cgi) that can quickly flood the wireless uplink and effectively bring GIAC off the internet. Line 4 disables NTP synchronization over the internet, an unnecessary service. CDP is disabled in line 5. Next, lines 6 and 7 disable IP unreachables and redirects. Unreachable packets can be used by an attacker to determine which IP addresses machines are responding at, helping narrow down the possible targets. Disabling them makes it appear as if every IP has a live machine behind it, making an attackers work slightly more difficult. Line 8 attaches access list 20 to ethernet 1. The last line disables SNMP and also returns the router to global config mode.

```
gw-giac(config)#int ethernet0
gw-giac(config-if)#ip address 200.200.200.1 255.255.255.0
        no ip directed-broadcast
        no cdp enable
        no ip unreachables
        no ip redirects
        ip access-group 50 in
        no snmp
gw-giac(config)#
```

The router is now brought into interface mode for ethernet 0, the connection from the router to the PIX. The second line gives ethernet 0 the first address in GIAC's space. Lines 3 through 6 specify the same settings as above, but ntp is left enabled. The router should properly be synced with the other machines on the internal network to keep log files and timestamps consistent; it simply does not need to sync off an internet NTP server. Line 7 attaches access list 50 to ethernet 0. Again, the last line disables SNMP and returns to global config mode.

```
gw-giac(config)#write memory
```

As a feature on Cisco products, all configuration commands only apply to the running software. If a mistake is made in configuration that blocks all access, one can simply power cycle the device to return it to its previous state. Running the above command writes the running configuration to memory so that it will be in place when the device is next rebooted. Forgetting to run this command now will wipe out all the settings that were put in place on the next power cycle.

Pix 525 – GIAC's firewall that protects its four networks:

The PIX is the heart of GIAC's security architecture. The router may protect GIAC from simple attacks from simple attackers, proxy servers may protect internal machines from malicious java servlets on web pages, switches may protect the network from network saturation attacks, and updating and patching may keep the servers' exposure limited, but without the firewall the entire network is up for grabs for any black hat who wants it. The policy for the PIX is fairly complicated, but will be simplified where it can be. The configuration of the PIX must keep these points in mind:

- Passthroughs must be provided for the web server (10.202.0.10), DNS, SMTP, Metaframe, and Webmail (10.202.0.7)
- DNS (10.202.0.2) requires using port 53/udp for everyone and 53/tcp for our secondary DNS server at Get Wireless, 199.199.199.12
- SMTP (10.202.0.3) requires inbound and outbound port 25/tcp
- Metaframe (10.201.0.2) requires inbound port 1494/tcp and 1494/udp
- Metaframe must be able to talk to the internet unrestricted
- Metaframe must be able to talk to an internal Novell server at 10.200.20.10 on port 524/tcp and other NT servers (10.200.20.2 and 10.200.20.3) on NetBIOS (ports 137-139, tcp and udp) for file and print sharing
- Metaframe needs to be able to talk to the logging server on the service network, 514/udp
- Metaframe needs to be able to access the internal web server (10.200.20.9) on port 80/tcp

© SANS Institute 2000 - 2005

Author retains full rights.

- Metaframe users will access Unix via ssh, port 22/tcp
- The internal web server requires access to the PostgreSQL server (10.202.0.5) on the service network at port 5432, tcp and udp
- The internal proxy server (10.200.20.5) needs access to the internet
- The sysadmin's station(s) (10.200.2.x) will need to be able to access the internet directly
- The sysadmin's station(s) will need to talk directly to the PIX and the router
- The sysadmin's station(s) will need to connect to Metaframe for administration
- Web developers need to be able to get to the machines hosting the app over NetBIOS, ports as above, from their server, 10.200.3.1
- Web developers need SSH access to the web and database servers, port 22/tcp, and access to the shares on NetBIOS, ports as above
- NetWare must be able to talk to clock.psu.edu (128.118.25.3) for NTP services
- All other servers on the service network and metaframe also need to talk NTP to clock.psu.edu. While these servers could easily synchronize with the NetWare server, the design of the network attempts to segregate the service and internal networks wherever possible
- NetWare will host GroupWise, which will need to access the SMTP server on the service network, port 25/tcp
- The internal DNS (10.200.20.7) will require access to DNS across the internet, port 53/udp
- The PIX must log to the internal logging server (10.200.20.8) and the service logging server (10.202.0.8)

With a list of requirements, a ruleset for the PIX may now be built. The purpose of this ruleset is to grant access for the services listed and deny all other traffic. This is the opposite of the router, where only specific traffic was denied and the remaining traffic was permitted. The configuration commands will be shown, followed by the config it generates, then with an explanation of it line by line. The syntax is for PIX v6.1, current as of this writing. Firmware will continue to be updated over the life of the device.

The Configuration Guide for the Cisco Secure PIX Firewall Version 6.0 (http://www.cisco.com/univercd/cc/td/doc/product/iaabu/pix/pix_60/config/config.htm) (forward compatible with later versions) provides two step-by-step tutorials, one to update the PIX software and the second to do initial configuration. The software is kept up to date by this method shortly after Cisco releases new versions – Cisco offers email announcements to alert of new releases. The GIAC PIX will run through the initial setup tutorial, entering commands in the global configuration mode:

```
enable
passwd <access password>
enable password <enable password>
config term
nameif ethernet0 outside security0
nameif ethernet1 inside security100
nameif ethernet2 metaframe security70
nameif ethernet3 service security60
ip address inside 10.200.0.1 255.255.0.0
ip address outside 200.200.200.2 255.255.255.128
ip address metaframe 10.201.0.1 255.255.255.128
ip address service 10.202.0.1 255.255.255.0
interface ethernet0 auto
interface ethernet1 auto
interface ethernet2 auto
interface ethernet3 auto
nat 1 0 0
global (outside) 1 200.200.200.3 netmask 255.255.255.255
global (metaframe) 1 10.201.0.50 netmask 255.255.255.255
global (service) 1 10.202.0.50 netmask 255.255.255.255
```

© SANS Institute 2000 - 2005 Author retains full rights.

```
route outside 0 0 200.200.200.1 1
static (inside,metaframe) 10.201.0.30 10.200.20.10
static (inside,metaframe) 10.201.0.22 10.200.20.2
static (inside,metaframe) 10.201.0.23 10.200.20.3
static (inside,metaframe) 10.201.0.20 10.200.20.20
static (inside,metaframe) 10.201.0.29 10.200.20.9
static (metaframe,outside) 200.200.200.10 10.201.0.2
static (service,outside) 200.200.200.12 10.202.0.2
static (service,outside) 200.200.200.13 10.202.0.3
static (service,outside) 200.200.200.17 10.202.0.7
static (service,outside) 200.200.200.20 10.202.0.10
static (inside,service) 10.202.0.20 10.202.0.9
static (inside,service) 10.202.0.31 10.200.3.1
access-list acl_metaframe permit icmp any any
access-list acl_service permit icmp any any
access-group acl_metaframe in interface metaframe
access-group acl_service in interface service
telnet 10.200.2.0 255.255.255.0 inside
telnet timeout 15
logging buffered errors
logging host service 10.202.0.8
logging host inside 10.200.20.8
logging on
logging timestamp
logging facility 3
```

This completes the statements from the PIX handbook. The firewall now has IP addresses, routes between interfaces, NAT enabled for all interfaces, and static mappings for some servers. The tutorial suggests enabling ICMP over DMZ devices, which will let GIAC test connectivity between networks much easier. Telnet service settings are set next, and last the logging settings are specified. The next step is to specify access lists for the different interfaces and the requirements of each along with some global commands not provided in the tutorial:

```
hostname giacpix
no rip inside passive
no rip inside default
no rip outside passive
no rip outside default
no rip service passive
no rip service default
no rip metaframe passive
no rip metaframe default
no snmp-server location
no snmp-server contact
snmp-server community caig-etavirp
access-list acl_out permit tcp any host 200.200.200.10 eq 1494
access-list acl_out permit udp any host 200.200.200.10 eq 1494
access-list acl_out permit udp any host 200.200.200.10 eq 1605
access-list acl_out permit tcp any host 200.200.200.10 eq 81
access-list acl_out permit tcp any host 200.200.200.13 eq 25
access-list acl_out permit tcp host 199.199.199.12 host 200.200.200.12 eq 53
access-list acl_out permit udp any host 200.200.200.12 eq 53
access-list acl_out permit udp host 128.118.25.3 any eq 123
access-list acl_out permit tcp any host 200.200.200.20 eq 80
access-list acl_out permit tcp any host 200.200.200.17 eq 80
```

© SANS Institute 2000 - 2005, Author retains full rights.

```
access-list acl_out deny ip any any
access-list acl_service permit udp host 10.202.0.2 any eq 53
access-list acl_service permit tcp host 10.202.0.3 any eq 25
access-list acl_service permit tcp host 10.202.0.10 any eq 80
access-list acl_service permit tcp host 10.202.0.7 any eq 80
access-list acl_service permit udp any host 128.118.25.3 eq 123
access-list acl_service permit tcp host 10.202.0.2 host 199.199.199.12 eq 53
access-list acl_service permit tcp host 10.202.0.5 10.202.0.31 255.255.255.0 eq 22
access-list acl_service permit udp host 10.202.0.5 10.202.0.31 255.255.255.0 eq 22
access-list acl_service permit tcp host 10.202.0.5 host 10.202.0.20 eq 5432
access-list acl_service permit udp host 10.202.0.5 host 10.202.0.20 eq 5432
access-list acl_service permit tcp host 10.202.0.10 host 10.202.0.31 eq 22
access-list acl_service permit udp host 10.202.0.10 host 10.202.0.31 eq 22
access-list acl_service permit tcp host 10.202.0.10 host 10.202.0.31 range 137 139
access-list acl_service permit udp host 10.202.0.10 host 10.202.0.31 range 137 139
access-list acl_service deny ip any any
access-list acl_metaframe permit tcp any any eq 1494
access-list acl_metaframe permit udp any any eq 1494
access-list acl_metaframe permit udp any any eq 1605
access-list acl_metaframe permit tcp any any eq 81
access-list acl_metaframe permit tcp any host 10.201.0.29 eq 80
access-list acl_metaframe permit tcp any host 10.201.0.30 eq 524
access-list acl_metaframe permit udp any host 10.201.0.30 eq 524
access-list acl_metaframe permit tcp any host 10.201.0.22 range 137 139
access-list acl_metaframe permit tcp any host 10.201.0.23 range 137 139
access-list acl_metaframe permit udp any host 10.201.0.22 range 137 139
access-list acl_metaframe permit udp any host 10.201.0.23 range 137 139
access-list acl_metaframe permit tcp any host 10.201.0.20 eq 22
access-list acl_metaframe permit udp any host 10.201.0.20 eq 22
access-list acl_metaframe permit ip any host 10.201.0.1
access-list acl_metaframe deny ip any 10.201.0.0 255.255.255.0
access-list acl_in permit udp host 10.200.20.7 any eq 53
access-list acl_in permit ip 10.200.2.0 255.255.255.0 any
access-list acl_in permit tcp 10.200.2.0 255.255.255.0 host 10.200.0.1 eq 23
access-list acl_in deny tcp 10.200.0.0 255.255.0.0 host 10.202.0.10 eq 80
access-list acl_in permit ip host 10.200.20.5 any
access-list acl_in permit tcp host 10.200.3.1 host 10.202.0.5 eq 22
access-list acl_in permit tcp host 10.200.3.1 host 10.202.0.10 eq 22
access-list acl_in permit udp host 10.200.3.1 host 10.202.0.5 eq 22
access-list acl_in permit udp host 10.200.3.1 host 10.202.0.10 eq 22
access-list acl_in permit tcp host 10.200.3.1 host 10.202.0.10 range 137 139
access-list acl_in permit udp host 10.200.3.1 host 10.202.0.10 range 137 139
access-list acl_in permit tcp host 10.200.20.9 host 10.202.0.5 eq 5432
access-list acl_in permit udp host 10.200.20.9 host 10.202.0.5 eq 5432
access-list acl_in permit tcp host 10.200.20.10 host 10.201.0.2 eq 524
access-list acl_in permit udp host 10.200.20.10 host 10.201.0.2 eq 524
access-list acl_in permit tcp host 10.200.20.10 host 10.202.0.3 eq 25
access-list acl_in permit udp host 10.200.20.10 host 128.118.25.3 eq 123
access-list acl_in permit tcp host 10.200.20.2 host 10.201.0.2 range 137 139
access-list acl_in permit tcp host 10.200.20.3 host 10.201.0.2 range 137 139
access-list acl_in permit udp host 10.200.20.2 host 10.201.0.2 range 137 139
access-list acl_in permit udp host 10.200.20.3 host 10.201.0.2 range 137 139
access-list acl_in deny ip any any
access-group acl_out in interface outside
```

Author retains full rights.

```
access-group acl_in in interface inside
write memory
```

Some global statements have been issued to restrict RIP traffic and to effectively disable SNMP, which is not currently being utilized. The rest of the statements are all access rules to enable or disallow different types of traffics. This completes the configuration of the PIX. The output of running “show config” on the firewall is shown in the appendix under “Firewall Config”. Sectioning the commands provides a better understanding of their usefulness.

Explanation of firewall config and tutorial:

Similar to the router, this tutorial will provide the user with the PIX commands that need run to enforce GIAC’s policy as well as a description of each command. Prompts will be shown when the user enters or exits a mode only. By logging onto the PIX, the user is in the base mode. First, enable mode must be entered and then config mode. The commands are:

```
pixfirewall> enable
Password:
pixfirewall# passwd <access password>
        enable password <enable password>
pixfirewall(config)# config term
```

The enable command lets a user make changes on the PIX. There is no password initially, so first an access password – the password an admin will enter when they first connect to the PIX – is added and then a password for the enable section. The last line, config term, enters the admin into configuration mode.

```
pixfirewall(config)# nameif ethernet0 outside security0
        nameif ethernet1 inside security100
        nameif ethernet2 metaframe security70
        nameif ethernet3 service security60
```

The nameif statements provide a logical name for each network interface on the firewall. The security numbers indicate the relative security of each interface, low being very secure and high being insecure. The outside interface is always the lowest security level, 0. Access from a higher security interface to a lower security interface, such as from inside to outside or metaframe to service, is allowed by default. Later access-list statements allow this policy to be changed. Access from a lower security interface to a higher security interface are disallowed and require the use of static commands, detailed later.

```
pixfirewall(config)# ip address inside 10.200.0.1 255.255.0.0
        ip address outside 200.200.200.2 255.255.255.128
        ip address metaframe 10.201.0.1 255.255.255.128
        ip address service 10.202.0.1 255.255.255.0
```

IP addresses are assigned to each interface with the IP scheme and netmask that were specified earlier. Devices on the three GIAC-controlled networks will point to the PIX for their default route, if they are to access the internet directly. For example, the DNS server will use 10.202.0.1 as its default route, but a desktop machine will not use the PIX as its default route.

```
pixfirewall(config)# interface ethernet0 auto
        interface ethernet1 auto
        interface ethernet2 auto
        interface ethernet3 auto
```

Each interface is set to auto-detect its speed. If problems with auto-negotiation show up, the speeds can be specified by using “10baset”, “10full”, “1000sxfull”, etc., instead of “auto”.

© SANS Institute 2000 - 2005 Author retains full rights.

```
pixfirewall(config)# nat 1 0 0
        global (outside) 1 200.200.200.3 netmask 255.255.255.255
        global (metaframe) 1 10.201.0.50 netmask 255.255.255.255
        global (service) 1 10.202.0.50 netmask 255.255.255.255
        route outside 0 0 200.200.200.1 1
```

Line one specifies that Network Address Translation, or NAT, is enabled across all interfaces. The next three lines specify the IP address that NAT will use on three of the interfaces. The last line provides the PIX with its default route to the internet, via Get Wireless's router.

NAT allows a client machine on one network, 10.200.0.8, to talk to a machine on another interface as if the PIX initiated the connection. This feature was historically enabled to conserve IP addresses – 5,000 users at one site could share one IP addresses for outgoing connections – but is more recently being used to aid security. All outgoing connections will "share" this one IP which will not accept incoming connections, thereby protecting the machines behind the firewall. In GIAC's case, NAT allows all machines accessing the internet to appear as if they originate at 200.200.200.3, all machines talking on the metaframe network as 10.201.0.50, and all machines talking on the service network as 10.202.0.50. NAT is only for connections that originate on another interface; two machines on the service network talking to each other will not be NAT'ed.

© SANS Institute 2000 - 2005, Author retains full rights.

```
pixfirewall(config)# static (inside,metaframe) 10.201.0.30 10.200.20.10
        static (inside,metaframe) 10.201.0.22 10.200.20.2
        static (inside,metaframe) 10.201.0.23 10.200.20.3
        static (inside,metaframe) 10.201.0.20 10.200.20.20
        static (inside,metaframe) 10.201.0.29 10.200.20.9
        static (metaframe,outside) 200.200.200.10 10.201.0.2
        static (service,outside) 200.200.200.12 10.202.0.2
        static (service,outside) 200.200.200.13 10.202.0.3
        static (service,outside) 200.200.200.17 10.202.0.7
        static (service,outside) 200.200.200.20 10.202.0.10
        static (inside,service) 10.202.0.20 10.202.0.9
        static (inside,service) 10.202.0.31 10.200.3.1
```

Because the PIX does not allow a lower security interface to talk to higher security interfaces directly, such as from outside to service, static commands are required to enable IP passthroughs. Each static statement matches an unused IP on the lower interface with a real IP on the higher interface. The first line establishes a static passthrough to the inside interface from the metaframe interface. It uses 10.201.0.21, an unused IP on the metaframe network, to indicate that traffic should then be passed to 10.200.20.1, a machine on the inside network.

First, a static command is set up for the NetWare server on the metaframe interface. Two NT servers are also set up on lines 2 and 3. The fourth line is for the SCO OpenServer box. Line 5 is the static for the internal web server. All are machines on the inside that Metaframe will need to talk to. Line 6 allows people on the internet to access the Metaframe machine at IP 200.200.200.10. The next four lines allow internet access to the DNS, SMTP, Webmail, and Web Access servers, respectively, on the service network. The next two lines are for servers on the inside that service network machines need to talk to. They are the internal web server and the web development server.

```
pixfirewall(config)# access-list acl_metaframe permit icmp any any
        access-list acl_service permit icmp any any
        access-group acl_metaframe in interface metaframe
        access-group acl_service in interface service
```

Access-list commands, like on the router, are what permit and deny packets. Unlike the router, however, the PIX is making its decisions based on the packet's type, its destination, and the destination port. Lines 1 and 2 allow all ICMP packets in and out of the metaframe and service networks. Lines 3 and 4 apply the lists to the appropriate interface. This is all done from the global command area; there is no equivalent to the router's interface specific mode. Similar commands are run later to fully describe the policy.

```
pixfirewall(config)# telnet 10.200.2.0 255.255.255.0 inside
        telnet timeout 15
```

Telnet access to the PIX itself is restricted to the machines on the sysadmin's network, 10.200.2.x. The timeout value in the second line disconnects idle sessions after 15 minutes.

```
pixfirewall(config)# logging buffered errors
        logging host service 10.202.0.8
        logging host inside 10.200.20.8
        logging on
        logging timestamp
        logging facility 3
```

Logging of the PIX is, of course, very important. First, logging is set to buffered and to the level "errors". This is the level of what messages will be seen when looking at the log on the console. The next two lines specify what servers should receive log statements. The fourth statement actually turns logging on, as the PIX comes with logging disabled. Line 5 specifies that timestamps should always be provided. It is particularly important to set the clock with a command such as

© SANS Institute 2000 - 2005 Author retains full rights.

clock set 21:00:00 jan 31 2002 or the timestamps will not match with other systems. The last command specifies the level of messages that will be sent to the logging servers; level 3 corresponds with errors. Higher levels, such as debugging, will quickly flood the logging server with useless information. The level can be set to debugging when drastic changes are made to the configuration, however.

```
pixfirewall(config)# hostname giacpix
giacpix(config)#
```

This command simply sets a hostname for the firewall.

```
giacpix(config)# no rip inside passive
        no rip inside default
        no rip outside passive
        no rip outside default
        no rip service passive
        no rip service default
        no rip metaframe passive
        no rip metaframe default
```

These commands disable RIP broadcasts and the PIX's ability to listen to other RIP broadcasts. They must be applied to each interface.

```
giacpix(config)# no snmp-server location
        no snmp-server contact
        snmp-server community caig-etavirp
```

The PIX will not broadcast any SNMP messages to any devices due to the first two lines. However, someone connecting to the PIX may still read SNMP messages off the PIX itself. The third line specifies the community as "caig-etavirp", or "private-giac" backwards. Popular SNMP community names are "public" and "private"; changing it to what looks like random letters decreases the possibility that someone can connect to view the SNMP data.

```
giacpix(config)# access-list acl_out permit tcp any host 200.200.200.10 eq 1494
        access-list acl_out permit udp any host 200.200.200.10 eq 1494
        access-list acl_out permit udp any host 200.200.200.10 eq 1605
        access-list acl_out permit tcp any host 200.200.200.10 eq 81
        access-list acl_out permit tcp any host 200.200.200.13 eq 25
        access-list acl_out permit tcp host 199.199.199.12 host 200.200.200.12 eq 53
        access-list acl_out permit udp any host 200.200.200.12 eq 53
        access-list acl_out permit udp host 128.118.25.3 any eq 123
        access-list acl_out permit tcp any host 200.200.200.20 eq 80
        access-list acl_out permit tcp any host 200.200.200.17 eq 80
        access-list acl_out deny ip any any
```

A limitation of the access-list command is that any statement specifying a port or port range – i.e. "eq 53" or "range 137 139" – cannot be used with the "ip" protocol, which specifies both tcp and udp protocols. Instead, two separate commands must be issued, one for udp and one for tcp. This adds some length to the access lists but is necessary.

The outside access list, acl_out, will test any incoming packets on the interface. Access lists on the PIX only test packets incoming to the interface; outgoing packets on an interface are not checked. To protect access to the service network, the other interfaces must be checked, starting with the outside. The PIX also only checks packets that cross the interface. When a packet moves from the inside network to the service network, NAT takes charge of the packet, treating it as if it does not cross the network. This makes guarding the networks somewhat simpler; a rule only needs to be applied on the inside interface against packets crossing to the service network, no return rule needs applied on the service interface.

© SANS Institute 2000 - 2005, Author retains full rights.

Packets coming across the internet are assumed to be intended for a publicly available GIAC service or are of malicious intent. Lines one and two allow packets to reach the Metaframe server, via its "static" IP on the outside network, and the third line permits Citrix Browsing on 1605/udp. The next line allows a user to enter "http://metaframe.giac.com:81" in a web browser and see applications published via NFuse. The NFuse apps are published on port 81 instead of 80 to reduce the number of web attacks thrown at the Metaframe server. The fifth line allows mail traffic to reach the SMTP server. The next two lines allow the DNS server to send zone transfers to *only* Get Wireless's DNS server and allows any other DNS queries to the server. Line eight allows traffic from clock.psu.edu on the time service to get through to internal machines. The following two lines allow http traffic to reach www.giac.com and webmail.giac.com.

The last line denies packets of any other types. The PIX has an implicit deny rule on any interface with an access-list applied to it. However, the implicit rule does not log dropped packets. By stating the deny rule, any packets that are denied are now logged. This rule will be applied to two of the other interfaces for the same reason.

This last deny rule should protect against the new widespread SNMP vulnerabilities (http://www.securityfocus.com/archive/1/255807) by blocking udp ports 161, 162, and 1993 as well as tcp port 1993.

```
giacpix(config)# access-list acl_service permit udp host 10.202.0.2 any eq 53
        access-list acl_service permit tcp host 10.202.0.3 any eq 25
        access-list acl_service permit tcp host 10.202.0.10 any eq 80
        access-list acl_service permit tcp host 10.202.0.7 any eq 80
        access-list acl_service permit udp any host 128.118.25.3 eq 123
        access-list acl_service permit tcp host 10.202.0.2 host 199.199.199.12 eq 53
        access-list acl_service permit tcp host 10.202.0.5 10.202.0.31 255.255.255.0 eq 22
        access-list acl_service permit udp host 10.202.0.5 10.202.0.31 255.255.255.0 eq 22
        access-list acl_service permit tcp host 10.202.0.5 host 10.202.0.20 eq 5432
        access-list acl_service permit udp host 10.202.0.5 host 10.202.0.20 eq 5432
        access-list acl_service permit tcp host 10.202.0.10 host 10.202.0.31 eq 22
        access-list acl_service permit udp host 10.202.0.10 host 10.202.0.31 eq 22
        access-list acl_service permit tcp host 10.202.0.10 host 10.202.0.31 range 137 139
        access-list acl_service permit udp host 10.202.0.10 host 10.202.0.31 range 137 139
        access-list acl_service deny ip any any
```

The next highest security interface is the service interface. By default, all access from a higher to lower interface is permitted, so an access list must be implemented to control access. First, standard DNS traffic is allowed. Mail traffic from the mail server is allowed out in line two. The web server and webmail also require outbound access on port 80, lines three and four. All hosts are then permitted to reach clock.psu.edu for time synchronization in line five. The sixth line allows the DNS server to send zone transfers to the secondary DNS server at Get Wireless, but no one else. Lines eight and nine allow ssh connections from the database server to the web development server. The next two rules allows the DB server to make PostgreSQL connections to the internal web server. SSH and NetBIOS connections from the web server to the web development server are permitted by the next four rules. The last line denies any other packet that was not permitted by the rules above.

```
giacpix(config)# access-list acl_metaframe permit tcp any any eq 1494
        access-list acl_metaframe permit udp any any eq 1494
        access-list acl_metaframe permit udp any any eq 1605
        access-list acl_metaframe permit tcp any any eq 81
        access-list acl_metaframe permit tcp any host 10.201.0.29 eq 80
        access-list acl_metaframe permit tcp any host 10.201.0.30 eq 524
        access-list acl_metaframe permit udp any host 10.201.0.30 eq 524
        access-list acl_metaframe permit tcp any host 10.201.0.22 range 137 139
        access-list acl_metaframe permit tcp any host 10.201.0.23 range 137 139
        access-list acl_metaframe permit udp any host 10.201.0.22 range 137 139
        access-list acl_metaframe permit udp any host 10.201.0.23 range 137 139
```

© SANS Institute 2000 - 2005

Author retains full rights.

```
access-list acl_metaframe permit tcp any host 10.201.0.20 eq 22
access-list acl_metaframe permit udp any host 10.201.0.20 eq 22
access-list acl_metaframe permit ip any host 10.201.0.1
access-list acl_metaframe deny ip any 10.201.0.0 255.255.255.0
```

Metaframe requires full access to the internet and partial access to the service network. Because the metaframe interface has higher access than both the service and outside interfaces, it is provided with full access to both. The question of whether access-lists should be applied to restrict access to the service network has no easy answer. One option is to add the access-lists, which then slows down the PIX as it has to pass all packets through a list of rules. The other option is to "trust" the Metaframe machine by not restricting it, requiring fewer rules on the firewall. The decision has been made to trust Metaframe. Strong password policies have been implemented and the only two ways to access it have been restricted as well (modem connections and only the ICA port on the internet). There is little reason to bog down the PIX with rules that will not likely be needed.

The first four rules allow ICA connections, Citrix browsing, and NFuse web applications to be used from anywhere. While Metaframe has unrestricted access to the service and outside interface, it has no access to the inside except via static commands; these statements allows outgoing ICA connections through the static addresses. The fifth line allows http access to the internal web server. Access is provided to the NetWare server and then both NT machines in the next six lines, five through ten. SSH access to the SCO OpenServer box is permitted in the 12th and 13th lines.

The last two lines enable unrestricted access to 10.201.0.1, the PIX and Metaframe's gateway, but deny all access to the rest of the 10.201.0.x network, effectively cutting off other access to the static IP's. Because the PIX works in a "first fit" instead of "best fit" mode, packets for the internet will be permitted by the 14th line rather than denied by the last line.

```
giacpix(config)# access-list acl_in permit udp host 10.200.20.7 any eq 53
access-list acl_in permit ip 10.200.2.0 255.255.255.0 any
access-list acl_in permit tcp 10.200.2.0 255.255.255.0 host 10.200.0.1 eq 23
access-list acl_in deny tcp 10.200.0.0 255.255.0.0 host 10.202.0.10 eq 80
access-list acl_in permit ip host 10.200.20.5 any
access-list acl_in permit tcp host 10.200.3.1 host 10.202.0.5 eq 22
access-list acl_in permit tcp host 10.200.3.1 host 10.202.0.10 eq 22
access-list acl_in permit udp host 10.200.3.1 host 10.202.0.5 eq 22
access-list acl_in permit udp host 10.200.3.1 host 10.202.0.10 eq 22
access-list acl_in permit tcp host 10.200.3.1 host 10.202.0.10 range 137 139
access-list acl_in permit udp host 10.200.3.1 host 10.202.0.10 range 137 139
access-list acl_in permit tcp host 10.200.20.9 host 10.202.0.5 eq 5432
access-list acl_in permit udp host 10.200.20.9 host 10.202.0.5 eq 5432
access-list acl_in permit tcp host 10.200.20.10 host 10.201.0.2 eq 524
access-list acl_in permit udp host 10.200.20.10 host 10.201.0.2 eq 524
access-list acl_in permit tcp host 10.200.20.10 host 10.202.0.3 eq 25
access-list acl_in permit udp host 10.200.20.10 host 128.118.25.3 eq 123
access-list acl_in permit tcp host 10.200.20.2 host 10.201.0.2 range 137 139
access-list acl_in permit tcp host 10.200.20.3 host 10.201.0.2 range 137 139
access-list acl_in permit udp host 10.200.20.2 host 10.201.0.2 range 137 139
access-list acl_in permit udp host 10.200.20.3 host 10.201.0.2 range 137 139
access-list acl_in deny ip any any
```

The last access-list, acl_in, manages packets from the internal network to any of the other networks. Access is permitted by default, but GIAC's policy is to deny access except where permitted. The rules are based on the premise of allowing specific access to the other three interfaces and then denying any other packets. This is similar to how acl_service was built, except the inside interface can talk to *all* other interfaces.

The first line allows the internal DNS server to make all the queries it needs to on behalf of internal machines. The administrator's machines are then allowed unfettered access to everything. The third line allows the administrator to

© SANS Institute 2000 - 2005

Author retains full rights.

telnet to the PIX, because of the previous line, but denies everyone else. Next, internal machines are denied access to www.giac.com – they are to use the internal web server only. The proxy server is now allowed unrestricted access in line 5, excepting the previous two denial statements. SSH access is then allowed from the web development server to the external web server and the database server, plus NetBIOS access to the web server, in the next four lines. The internal web server is then allowed to make PostgreSQL connections to the database server in lines 12 and 13. On lines 14-17, NetWare is allowed to communicate with Metaframe, send mail via the SMTP server, and synchronize time with clock.psu.edu. Lines 19 through 22 allow the two NT servers to share files with Metaframe. The last line denies all other traffic.

```
giacpix(config)# write memory
```

Last, but not least, the changes need to be written to memory.

The ruleset, as a whole, provides the availability to the required services without unduly compromising security. The firewall is an effective traffic manager and greatly augments the security of GIAC's networks. However, it is not a 100% guarantee of valid traffic. The services being used – including DNS, SMTP, SSH, and web access to customers – must themselves be secured.

VPN - Metaframe Remote Access:

Metaframe was chosen to provide remote access for a number of reasons. Primary was the ability to secure it as part of the firewall policies rather than another layer of extensive work on the server. Secondary was its ability to be functional if GIAC's internet access fails, via modems.

The firewall in front of the PIX secures the Metaframe server, running Windows 2000, from all traffic except incoming and outgoing ICA connections (port 1494, tcp and udp), Citrix browsing (1605/udp) and outgoing DNS queries (53/udp). This is not an excuse to not install service packs and hotfixes, but enables the administrators to do their work rather than worrying about which default services must be disabled. With very stringent password policies and some changes to the filesystem permissions the server can be kept secure with very little hassle.

Users who wish to have access to remote access under Metaframe must ask for it and have their department head's authorization. Local users are then created on Metaframe; it uses its own user database for initial access regardless of what users might be on the NT and NetWare servers it can connect to. Users are created with default policies of 30-day password rotation which are then scanned with the John the Ripper password cracking utility (http://www.openwall.com/john/). If passwords are cracked within 30 days, the user's access is disabled until their password is changed. Passwords must contain 8 characters with a minimum of two capital letters, two digits, and one special character (semi-colon, exclamation mark, etc.) A password such as "!42inSEF" is excellent, but a user with a password of "qwerty123" would very quickly have their access revoked.

A login script is also specified for most users. Each department will have a login script created for it, such as "hr.bat" or "controllers.bat", for drive mappings to the internal NT or NetWare servers. The home directories will reside on the Metaframe root drive, M:, where they can be backed up easily. While the home directory is more likely to be used by programs to store settings than by the user themselves, the directories will be locked down to prevent more intelligent users from damaging theirs or others settings. The correct permissions for a home directory is "Full" access for Administrators and "Change" access for the user, removing all access for the Everyone group (enabled by default). Figures 2.1 and 2.2 show the user named "rdnelson" being created and the file permissions being set on his home directory.

© SANS Institute 2000 - 2005, Author retains full rights.

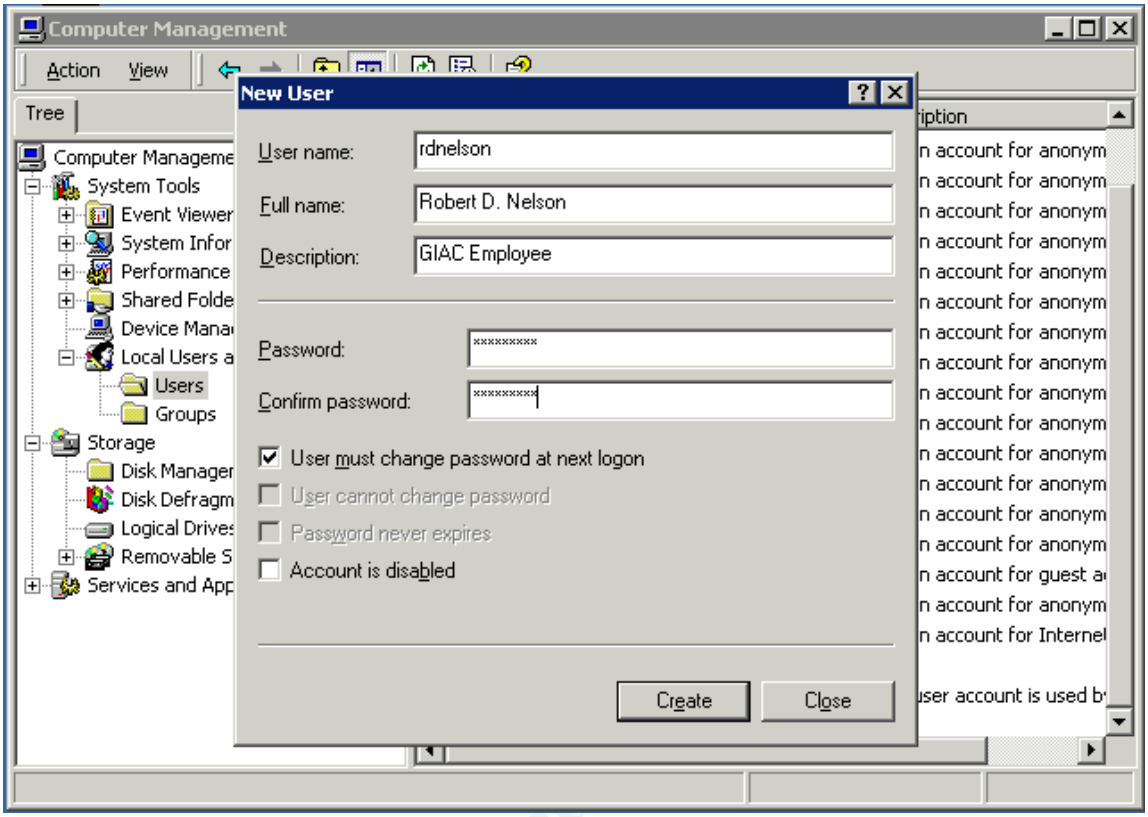


Figure 2.1 - Creating a local user

© SANS Institute 2000 - 2005

Author retains full rights.

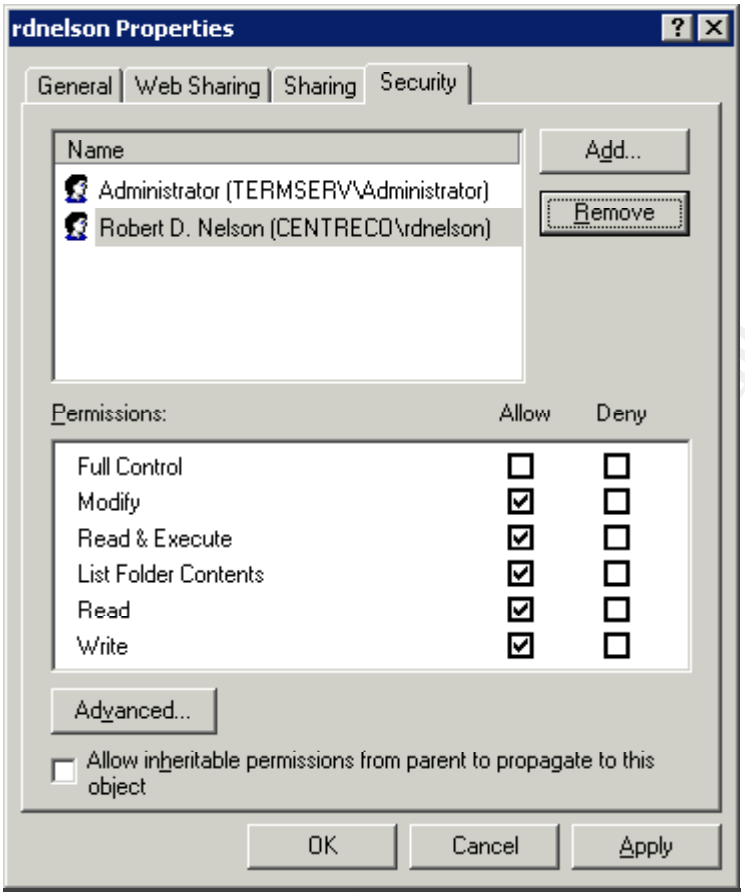


Figure 2.2 – Setting permissions on the home directory

During the Metaframe install, the choice for filesystem permissions compatible with Windows 2000 users will be made. This provides default filesystem permissions locking down the registry and important parts of the filesystem, such as C:\WINNT and C:\Program Files, so that non-privileged users have the ability to run programs but not modify. A user may potentially upload a file from his or her laptop, install it in their directory-space, and disable their own ability to use Metaframe, but at no point will they impact anyone else's ability to use the box. The one exception to this is loading and running programs that overload the hardware. The only policy that can limit this, however, is the Employee's Manual and Computer Policy Handbook. Users violating this policy, knowingly or unknowingly, will have their access to Metaframe removed for a full 30 days on first violation. Further violations will be handled by the Human Resources department as they deem necessary, accompanied with increasingly long periods of revoked access.

To secure the connections themselves, the ica-tcp and ica-com*N* connections must be modified. No access will be granted on a modem or internet connection without proper user credentials, but leaving disconnected sessions open without resetting them can be a drag on system resources, a low-tech denial of service not necessarily of ill intent. Connections are set on the ica-tcp connection, as in Figure 2-3, to disconnect after 10 minutes of idle time and to then reset disconnected sessions after 5 minutes. Also, other settings such as client drive/printer mappings are set here, as shown in the figure. The settings for the modems, ica-com1 through ica-com*N*, are set in the same way.

© SANS Institute 2000 - 2005

Author retains full rights.

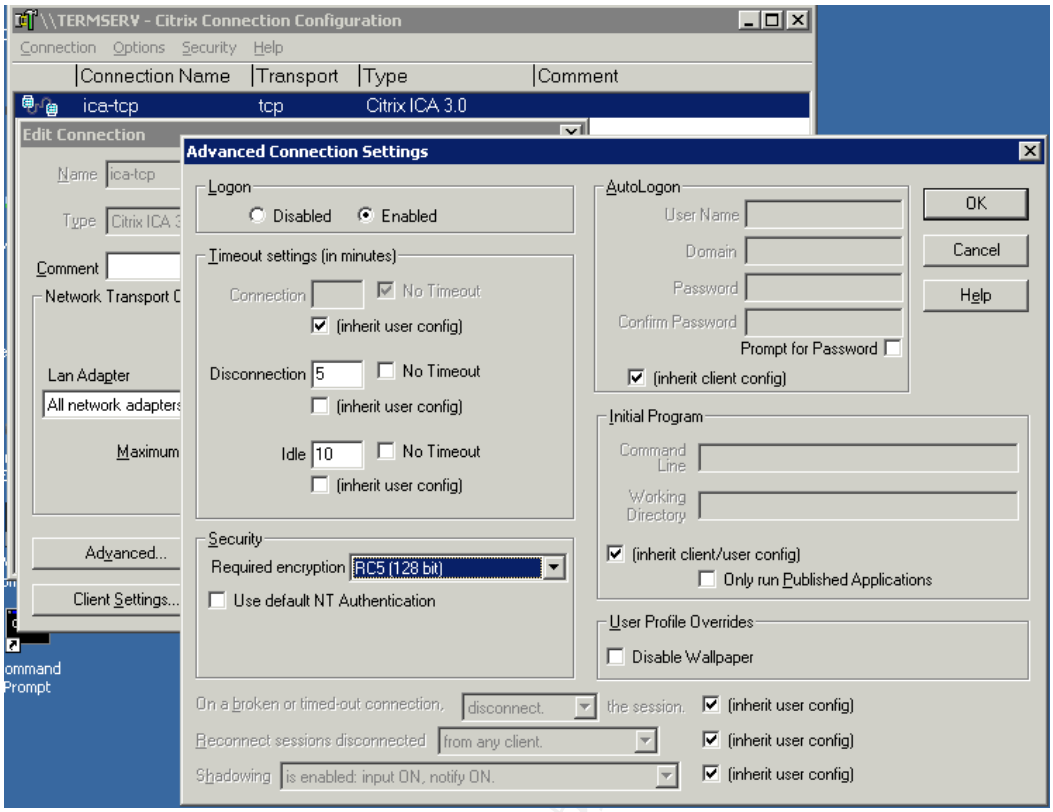


Figure 2.3 – Setting ICA-TCP connection's disconnect and idle timeouts

There is little else that can be done to secure Metaframe that the Firewall doesn't already prevent. User directories are protected, password policies and auditing is enabled, and idle-session DoS's are prevented on Metaframe itself. Attacks on other vulnerable services are prevented by the firewall and through the regular application of patches. A policy is also in place to monitor the users for inappropriate use of facilities as well.

Notes:

Some of the policies above are very stringent in specifying the exact details of connections that can or can't work. However, paranoia is very important in designing a secure network. As policies tend to relax over time rather than tighten up, it is often better to make the initial policy very tight instead of planning to tighten it later.

Also, the ports specified in the PIX's access lists are according to the Internet Assigned Numbers Authority "Well Known Port Numbers" (http://www.iana.org/assignments/port-numbers), considered the authoritative source on service ports. Vendors frequently provide incomplete lists of ports used by their products, leading to frustrations later when the problem is discovered at 2:12 on a Sunday morning. Specifically, Citrix documentation states that only port 1494 protocol tcp is required to be used through the firewall, but if udp is not enabled then certain functions do not work.

Testing:

Each piece of the access lists and rules above must be checked. Regular audits will prove how the overall policy works, but it is important to have an understanding of how to test individual parts.

A few of the rules on the firewall can be tested quickly with telnet and nmap. The rules being tested are:

```
access-list acl_service permit tcp host 10.202.0.3 any eq 25
access-list acl_metaframe permit tcp any host 10.201.0.29 eq 80
access-list acl_in deny tcp 10.200.0.0 255.255.0.0 host 10.202.0.10 eq 80
```

© SANS Institute 2000 - 2005

Author retains full rights.

The three access list statements allow access to the mailserver from anywhere, to the internal web server port 80/tcp from metaframe, and denies access from internal machines to the external mail server.

By logging on at the Metaframe server, the first two rules can be checked. Telneting to ports 25 on 10.202.0.3 and port 80 on 10.201.0.29 will suffice. In the first case, the user should expect to see an SMTP banner, as below. When the user telnets to port 80 and types “get / html/1.1”, they should get an HTTP error code as shown in the second section below. Telneting to port 80 on the mailserver or port 25 on the webserver should fail. At this time, an administrator can check the log files of the pix and find an entry similar to that shown in the third section below.

```
C:\>telnet 10.202.0.3 25
220 mail.co.centre.pa.us InterChange ESMTP v3.61.01 Ready

C:\telnet 10.201.0.29 80
HTTP/1.1 400 Bad Request
Server: Microsoft-IIS/5.0
Date: Fri, 22 Feb 2002 21:12:37 GMT
...

C:\telnet 10.201.0.29 25
Connecting To 10.201.0.29...Could not open a connection to host on port 25 : Connect failed

logging# tail /var/log/pixfirewall
Feb 22 16:28:57 pix %PIX-4-106023: Deny tcp src metaframe:10.201.0.2/1393 dst inside:10.200.20.9/25 by
access-group "acl_metaframe"
```

The last rule can be checked from an internal machine with a web browser. Using a proxy address of 10.200.0.5, the user should be able to browse the web. Disabling the proxy should stop the user from being able to browse. This can be verified by looking at the logs again:

```
logging# tail /var/log/pixfirewall
Feb 22 16:35:27 pix %PIX-4-106023: Deny tcp src inside:10.200.0.85/3201 dst outside:19.24.174.117/80 by
access-group "acl_in"
```

Other individual rules can be checked in the same process if they are added one at a time. However, a large number of rules added at once or a global check of all the policy statements is best done by a comprehensive audit. The methodology described here can be applied to every step of the audit, making the skill-set requirements of an auditor fairly low.

Summary:

The prime vulnerabilities of a network are its connection to other networks that are not trusted. GIAC’s main points of failure are the border router provided by Get Wireless, GIAC’s own PIX 525 firewall, and remote access via Metaframe. The router is secured by an agreement between GIAC and Get Wireless that allows GIAC to enforce its policies at the border. The PIX contains a very thorough ruleset, encompassing the policies of three different networks’ interactions as well as their combined access to the internet. Remote access is secured mostly through the PIX but also on the Metaframe server itself. Protection is in place against known malicious attacks; logging and intrusion detection have been set up to provide for new and unknown attacks. The devices are all very flexible, allowing both GIAC’s security administrators to continue to provide the same level of service against future security issues and GIAC to have a high return on investment.

The router, firewall, Metaframe server, and other network devices will be updated constantly to keep up with new developments in the security community. The primary method for updating will be through audits that check to ensure the current level of security as well as to check against new vulnerabilities. The next section will describe the audits in full.

© SANS Institute 2000 - 2005 Author retains full rights.

Assignment 3 – Audit Your Security Architecture (25 points) You have been asked to conduct a technical audit of the primary firewall (described in Assignments 1 and 2) for GIAC Enterprises. In order to conduct the audit, you will need to:

1. *Plan the audit. Describe the technical approach you recommend to assess the firewall. Be certain to include considerations such as what shift or day you would do the assessment. Estimate costs and level of effort. Identify risks and considerations.*
2. *Conduct the audit. Using the approach you described, validate that the primary firewall is actually implementing GIAC Enterprises' security policy. Be certain to state exactly how you do this, including the tools and commands used. Include screen shots in your report if possible.*
3. *Evaluate the audit. Based on your assessment (and referring to data from your assessment), analyze the perimeter defense and make recommendations for improvements or alternate architectures. Diagrams are strongly recommended for this part of the assignment.*

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

© SANS Institute 2000 - 2005, Author retains full rights.

Section 3 Layout

1. Auditing the firewall (“You have been asked to conduct a technical audit of the primary firewall”)

2. Planning (“Plan the audit”)

   A. Describe timing of event, costs, and permission that needs to be obtained (“Be certain to include considerations such as what shift or day you would do the assessment. Estimate costs and level of effort. Identify risks and considerations”)
   B. Describe the tests to be run and a checklist of services and protocols to be verified (“Describe the technical approach you recommend to assess the firewall”)
   C. Include a list of tools and commands that will be used as well as required parameters and usage examples (“Be certain to state exactly how you do this, including the tools and commands used.”)

3. Performing the audit (“Conduct the audit”)

   A. Perform the pre-defined tests while monitoring logs (“Using the approach you described, validate that the primary firewall is actually implementing GIAC Enterprises’ security policy”)
   B. Provide results of tests as in-line text rather than screen shots (“Include screen shots in your report if possible”)

4. Results of the audit (“Evaluate the audit”)

   A. Provide the results of the audit; what passed and what failed? (“Based on your assessment (and referring to data from your assessment)” and “Include screen shots in your report if possible”)
   B. Provide assessment of the effectiveness of the firewall in enforcing the policies (“analyze the perimeter defense”)
   C. Outline where the firewall failed and what steps can be taken to remedy this (“make recommendations for improvements” and “Diagrams are strongly recommended for this part of the assignment”)
   D. Propose alternative architecture where appropriate, including caveats (“make recommendations for … alternate architectures”)

5. Summary

© SANS Institute 2000 - 2005 Author retains full rights.

Section 3: Policy Audits

Auditing the Firewall:

The firewall provides most of the protection to the network, including the service network and the metaframe network. It must be kept up to date with newfound security holes and prevention techniques. A regular audit on a 6-month interval must be done. Additionally, quarterly audits of small segments, such as probing only the Metaframe server, should also be scheduled to help prevent lax security. The primary 6-month audit is documented below, from the planning stage through the execution and the evaluation of the results.

Planning an audit:

An audit only requires a few additional freeware tools in software but will require a few technicians or system administrators to perform the audit. Because GIAC works with many partners worldwide and customers have access to the GIAC webpage 24 hours a day, there is no perfect time to run an audit. To minimize the potential disruption, the audit will be performed at 8pm on a Saturday evening. Companies on the west coast will be closing for the day and those performing the audit will have had time to eat dinner. Companies in other countries further west, such as Japan, Russia, and Hong Kong, will be waking up to a Sunday morning, a very unlikely time for them to require access to GIAC's services. Europe, Africa, and western Asia will all be preparing for bed. By performing all audits – the 6 month audit and the quarterly audits – on Saturday evenings, GIAC will be able to disrupt the network for audits without significantly affecting GIAC's partners, suppliers, or customers. Additionally, if a problem is discovered, the auditors have 36 hours to fix it before GIAC opens again on Monday.

The cost of the audit is very low. The tools (listed below) are all free. The administrators performing the audit will be "paid" with comp time. No special hardware needs purchased; the laptops used can be any two laptops from among the administrators' group. The audits are expected to last no longer than 2 hours, although the first audit will probably run longer. A few hours of prep time on Friday will be required as well. The total "cost" in man-hours should be 6 hours (1 hr. prep., 2 hr. for the audit, 2 auditors) and $0 in hardware and software. High-end estimates, if a serious problem is encountered, is a total of 12 man-hours – the initial 6 hours plus an extra 3 hours per auditor to remedy the problem.

The overall effort put into the audit is fairly low. The admins need to boot a few laptops and carry them around, plugging and unplugging network jacks. The admins must also work pretty fast if they want to finish by 10pm. However, if there is a significant problem with the audit or with one of the servers while the admins are there, they can expend to be working a lot harder to get the problem under control and fixed.

If a problem gets out of hand or the administrators need a hand from a colleague for a particular problem, the audit may turn into a very long ordeal. This is a risk that need be taken, however. The risk itself is fairly low – none of the hardware is being rebooted and no configurations are being tampered with – but these things do happen. The auditors should try and plan the audit so that most of the other administrators are at least within the local area over the weekend. Planning the audit while the company guru is in Hawaii on vacation would not be a smart move.

Management at GIAC is very flexible about weekend work. The auditors are to send a registered email to the IT director and their direct boss (if any) to let them know about the proposed downtime. The IT director is expected to notify the appropriate people (both internal and external) and to let the auditors know if there is a problem with the proposed time. By sending the email on Monday, the auditors can perform their job on Saturday evening with reasonable certainty that the disruption won't affect GIAC.

To perform an audit, the policy requirements listed for the firewall in Section 2 need to be enumerated into services by the ports and protocols they use. The protocol type ip denotes both tcp and udp access on a port. The list of the current audit points is:

Incoming on the outside interface

- Deny RFC1918 private addresses

SANS Institute 2000 - 2005, Author retains full rights.

- Allow 25/tcp to 200.200.200.13
- Allow 53/udp to 200.200.200.12
- Allow 53/tcp to 200.200.200.12 from 199.199.199.12 only
- Allow 1494/ip to 200.200.200.10
- Allow 1605/udp to 200.200.200.10
- Allow 81/tcp to 200.200.200.10
- Allow 80/tcp to 200.200.200.17 and 200.200.200.20
- Allow 123/udp from clock.psu.edu
- Deny any other packets

Incoming on the service network

- Allow 37/ip to clock.psu.edu
- Allow 53/udp from 10.202.0.2 to any
- Allow 53/tcp from 10.202.0.2 to 199.199.199.12
- Allow 25/tcp from 10.202.0.3 to any
- Allow 22/ip from 10.202.0.5 to 10.202.0.31
- Allow 5432/ip from 10.202.0.5 to 10.202.0.20
- Allow 80/tcp from 10.202.0.7 to any
- Allow 137-139/ip from 10.202.0.10 to 10.202.0.31
- Allow 22/ip from 10.202.0.10 to 10.202.0.31
- Allow 80/tcp from 10.202.0.10 to any
- Deny any other packets

Incoming on the metaframe interface

- Allow 123/udp to clock.psu.edu
- Allow 514/udp to 10.202.0.8
- Allow 53/udp to 10.202.0.2
- Allow 80/tcp to 10.201.0.29
- Allow 81/tcp to any
- Allow 524/ip to 10.201.0.30
- Allow 137-139/ip to 10.201.0.22 and 10.201.0.23
- Allow 22/ip to 10.201.0.20
- Allow 1494/ip to all
- Allow 1605/udp to all
- Allow all from 10.201.0.2 to the internet
- Deny any other packets

Incoming on the inside interface

- Allow all from 10.200.2.x to any
- Allow 23/tcp (telnet) from 10.200.2.x to 10.200.0.1
- Deny 23/tcp from any other host to 10.200.0.1
- Allow 137-139/ip from 10.200.3.1 to 10.202.0.10
- Allow 22/ip from 10.200.3.1 to 10.202.0.10
- Allow 22/ip from 10.200.3.1 to 10.202.0.5
- Allow 53/udp from 10.200.20.7 to any
- Allow 524/ip from 10.200.20.10 to 10.201.0.2
- Allow 123/udp from 10.200.20.10 to clock.psu.edu
- Allow 25/tcp from 10.200.20.10 to 10.202.0.3
- Allow 137-139/ip from 10.200.20.2 and 10.200.20.3 to 10.201.0.2
- Allow 5432/ip from 10.200.20.9 to 10.202.0.5

© SANS Institute 2000 - 2005, Author retains full rights.

- Allow all from 10.200.20.5 to any
- Deny any other packets

This is the same policy as was described before but reduced to port access and IP addresses, the "language" that auditing tools speak.
To perform audit, a tool named nmap will be used (NMAP Stealth Port Scanner by Fyodor, http://www.insecure.org/nmap). Nmap allows a user to probe a system based on protocols, ports, and IP range, gathering a variety of information on the data specified as well as the target system itself. For instance, it can scan UDP ports 1, 3, 57, and 120-1999 on IP addresses 10.200.2.x, 193.23.53.1-10, and 43.9.23.153, or it can scan TCP ports 50-100 on a router at 65.9.102.1 and determine who makes it and what version of firmware it uses. This is the primary tool that will be used for determining if traffic is allowed or denied on the listed ports.

However, nmap is restricted in that it requires some sort of response from the target system. To check to see if all packets are denied on a port that a machine doesn't even service, nmap cannot give a valid answer – it could be filtered (deny statement) or closed (machine doesn't service the port). To assist, the PIX logs can be analyzed to see what logging messages are reported. The PIX also displays the number of times a rule has been matched when the command "show access-list <acl name>" is given. Except on the busiest interface, outside, checking the rule match numbers can show which rule was used to permit or deny a single packet.

Nmap and logs are, sometimes, a fairly high tech solution to some of the simple solutions. The utility "ping" is good for determining whether ICMP information is being passed and using the services on the network, such as Metaframe or the webmail server, are definite proofs that the services are or aren't being allowed. Even if packets are being passed, it is no guarantee that the service *works* in the intended manner.

As a last resort, if nmap, logs, or client programs can't determine the status of a service, tcpdump can be used. Tcpdump and its Windows analog Windump (www.tcpdump.org and www.windump.org, respectively) are packet sniffers that would allow an auditor to watch packets on the network to see whether they get through an interface or are stopped. By running one copy of tcpdump on either end, the auditor should see a packet inbound on one interface and outbound on the other for a permitted packet or inbound only if the packet is denied. All of the ports and services in GIAC's current setup can be determined without the aid of tcpdump, but for troubleshooting and possible future setups it is useful.

The auditing can now be done with two people. One person on an admin station, a computer with an ip address in the 10.200.2.x network, can monitor the logs on the PIX itself and on the internal logging server. The other auditor will use two laptops: one running Linux with nmap, tcpdump, and a web browser; the other running Windows 2000 with the Citrix Metaframe client, windump, and a web browser. By plugging the laptops into a hub alongside the PIX interface on each network – such as on a hub that contains Metaframe, the laptop, and the metaframe PIX interface – nmap and other tools can probe the PIX's policy for that interface. GIAC will keep at least two 4 port hubs handy for auditing, one for the interface being audited and the second in case tcpdump is required on another interface for more testing. As this is being done on a Saturday evening, there should be very little disruption in traffic if the auditors have to plug and unplug the PIX from the different interfaces repeatedly.

Tool	Command Form	Typical options	Description of options
Nmap	nmap [scantype] [options] <target> [secondary options]	-sU, -sS, -P0 [options] -T5 [secondary options]	UDP scan, SYN scan, do not ping target first, use .4s timeout value
Ping	ping <target IP> -t	-t	Continuous ping
Tcpdump	tcpdump [options] -i <interface> [filter]	-nn [options] port XX [filter]	Do not do name or port resolution, only show packets to/from port XX
Windump	windump [options] <interface> [filter]	-nn [options] port XX [filter]	Do not do name or port resolution, only show packets to/from port XX
telnet	telnet <target IP> [port]	25, 53 [ports]	Common ports for mail and DNS
Nslookup	nslookup [type] <target>	type=mx	Look up Mail eXchange records
log files	tail <file> [options]	-f	"follow" a log file by continuously updating the screen with log entries

© SANS Institute 2000 - 2005

Author retains full rights.

Performing the audit:

The first step in the audit is to start monitoring the logs. By logging into the PIX and entering enable mode, the first auditor can run “show access-list <acl name>” to view the number of matches on each rule. Opening a telnet session to 10.200.20.8 and tailing the log file, such as “tail –f /var/log/pixfirewall”, will display the PIX logs in real-time. The auditor will do a lot of standing around and waiting initially, so he should help his partner with setting up the laptops when log monitoring isn’t needed – both auditors will get home a lot quicker.

```
giacfw# show access-list acl_in
access-list acl_in permit udp host 10.200.20.7 any eq 53 (hitcnt=3028)
access-list acl_in permit ip 10.200.2.0 255.255.255.0 any (hitcnt=103)
...

logging# tail /var/log/pixfirewall –t
Feb 20 10:18:57 pix %PIX-4-106023: Deny tcp src outside:199.234.153.64/4493 dst
inside:199.234.154.117/113 by access-group "acl_out"
Feb 20 10:19:19 pix %PIX-4-106023: Deny udp src inside:10.200.0.4/137 dst outside:172.17.40.255/137 by
access-group "acl_in"
...
```

The second auditor can now hook up the Windows laptop to the outside interface. By providing it with the IP of 200.200.200.200, it will be on the same network as the router and firewall but far enough removed from the firewall that it won’t interrupt the static IP’s. The auditor will open a web browser on the laptop. He should be able to connect to www.giac.com and webmail.giac.com without a problem. He should also be able to get to http://metaframe.giac.com:81 and see a list of Citrix published applications after authenticating with a valid user. Next, he will open the Citrix client and set up a TCP/IP connection to metaframe.giac.com, which should resolve to 200.200.200.10. The connection should prompt for authentication and then allow him to see the desktop of the user used. Now, 4 rules – 1494/ip, 1605udp, 81/tcp, and 80/tcp – have been validated to work.

The DNS server of the laptop will now be set to 200.200.200.12, GIAC’s primary DNS server. Bringing up a command prompt, the auditor will run nslookup. Nslookup will do name lookups on IP’s and names. The DNS server, which resides on the service network, should provide answers for entries like webmail.giac.com and metaframe.giac.com and their IP counterparts, such as 200.200.200.12, as well as provide lookups for the other machines on the service network and Metaframe. To make sure that only Get Wireless can request and receive DNS zone transfers, the auditor enters “ls –d giac.com” into nslookup. If a complete listing of the domain is provided, starting with the SOA record and proceeding through all the A and PTR records, then anyone can view zone transfers. The results of running this command against 200.200.200.12 are shown below. Nslookup can now be closed.

```
> server 200.200.200.13
Default Server:  dns.giac.com
Address:  200.200.200.13

> ls -d co.centre.pa.us
* request timed out
```

The next rule to test is port 25/tcp access. By running the command “telnet mail.giac.com 25” the laptop should connect to the mailserver, seeing the prompt shown below. If the allow rule had failed, telnet would not have been able to connect to the host. The auditor can now switch to the Linux laptop and take the windows laptop off the hub.

```
[ronelson@lankhmar ronelson]$ telnet mail.giac.com 25
Trying 200.200.200.13...
Connected to mail.giac.com.
Escape character is '^]'.
```

© SANS Institute 2000 - 2005 Author retains full rights.

© SANS Institute 2000 - 2005 Author retains full rights.

```
220 ********************************************0*******
helo ronelson.giac.com
250 Ok, hello ronelson.giac.com.
```

The denial policy can easily be tested with nmap from the Linux laptop. Running nmap against any of the static IP's, such as 200.200.200.17, for ports 1-1,023 should show all ports closed except those which have an explicit policy to allow them, such as port 80 in this case. A scan of all ports, 1 through 65535, should provide the same results, but the example here shows only 1,023 ports for brevity of testing speed. Scanning all the ports on a machine can take quite a while, especially on ports where the service can expect a large timeout value. The results of running nmap against the mailserver with a tcp and a udp scan are shown below. To effectively scan the whole range of ports, the auditors will run "nmap [–sU] –P0 –p1-1023 200.200.200.1-20 –T5". The option –P0 is important because it tells nmap not to ping the host first; since no ICMP pings are allowed on the outside interface, nmap's default behavior is to not scan unpingable hosts. The default scan mode is TCP; running nmap a second time with –sU will do a UDP scan. Connection attempts can take a long time to time-out for filtered ports, so using –T5 tells nmap to be very aggressive and only try each port for .4 seconds. The results of the nmap scan will only show the few ports listed in the beginning of this section under "Incoming on the outside interface" as open; the mailserver scan below should only show port 25/tcp as open.

```
[root@lankhmar /root]# nmap -P0 -p1-1023 mail.giac.com -T5

Starting nmap V. 2.54BETA22 ( www.insecure.org/nmap/ )
Interesting ports on mail.giac.com (200.200.200.13):
(The 1022 ports scanned but not shown below are in state: filtered)
Port       State       Service
25/tcp     open        smtp


Nmap run completed -- 1 IP address (1 host up) scanned in 1400 seconds

[root@lankhmar /root]# nmap -sU -P0 -p1-1023 mail.giac.com -T5

Starting nmap V. 2.54BETA22 ( www.insecure.org/nmap/ )
All 1023 scanned ports on mail.giac.com (200.200.200.13) are: filtered

Nmap run completed -- 1 IP address (1 host up) scanned in 60 seconds
```

GIAC's PIX is also supposed to block RFC1918 addresses. To verify this, nmap will be used again. The option "-S <address>" alters the source IP to the address specified. Nmap will not be able to spoof the source address with the regular scan, so the option "-sS", a SYN scan, must be used. By specifying addresses of 10.x.x.x, 192.168.x.x, and 172.16.x.x, the auditors can see if packets can get to ports that are otherwise enabled. Using the mailserver again, the command "nmap –P0 -sS –S 10.0.0.5 –p 25 mail.giac.org" should report that port 25 is filtered or closed rather than open. The audit should also be performed with a 192.168 and a 172.16 address. The results of scanning from 10.0.0.5 are shown below.

```
[root@lankhmar /root]# nmap -sS -S 10.0.0.5 -P0 –p1-50 mail.giac.com -T5

Starting nmap V. 2.54BETA22 ( www.insecure.org/nmap/ )
All 50 scanned ports on mail.giac.com (200.200.200.13) are: filtered

Nmap run completed -- 1 IP address (1 host up) scanned in 6 seconds
```

The last rule, allow NTP (123/udp) to all, can not be adequately tested on the outside network, as connections to clock.psu.edu will be initiated from other interfaces. However, without the rule, return traffic for the connection would be denied. This will be tested separately on each interface. Likewise, some of the policies on the other interfaces have been proven effective by the testing on the outside. They will be discussed in each section.

© SANS Institute 2000 - 2005, Author retains full rights.

The auditors will also want to check some random ports they know are open on the servers that should be blocked, such as port 3389/tcp for Windows 2000 Terminal Services, by trying to connect to the services. First, the auditors set the IP of the laptop back to 200.200.200.200. Using the terminal service client and setting it to try and connect to 200.200.200.17, the Webmail server running Windows 2000, will try to open a connection on port 3389/tcp. The auditors never get a login screen. They verify that the port was blocked, rather than simply not responding, by looking at the logs. The last entry in the log file /var/log/pixfirewall is now:

> Feb 20 20:19:19 pix %PIX-4-106023: Deny tcp src outside:200.200.200.200/4011 dst service:10.202.0.7/3389 by access-group "acl_out"

Other ports, such as Windows file sharing and directory services, will be tested by the auditors for completeness. The scans will not be shown here, but the auditors are to try at least five different services – they choose which ones to test – as part of the audit.

The auditor will now move the Linux laptop and hub to the service network, giving it IP address 10.202.0.30. Some of the policies here have already been proven on the outside interface. Allowing 53/udp outbound and 25/tcp has been fully proven, allowing 80/tcp from the webserver and webmail has been proven accessible from the outside but not the other networks.

The first issue is testing the ability to send zone transfers from GIAC's DNS server to Get Wireless's DNS at 199.199.199.12. This can be tested by running "telnet 199.199.199.12 53" on the laptop and on the DNS server. The DNS server should get a prompt and the laptop should not be able to connect. To prove that the rule adequately allows access to do zone transfers, the DNS admin can make a minor change in the DNS records a few days before the audit, perhaps adding a bogus record like audit.giac.com, and attempting to read that entry off of Get Wireless's server. Firing up nslookup again, the admin will enter two commands. One is "server 199.199.199.12", changing the server to Get Wireless's. Next, he will enter "audit.giac.com". If the record is available, then zone transfers work. An example is shown below:

```
giacaudit# nslookup
Default Server: dns.giac.com
Address:  200.200.200.12

>server 199.199.199.12
Default Server: ns1.getwireless.net
Address: 199.199.199.12

>audit.giac.com
Server: ns1.getwireless.net
Address: 199.199.199.12

Name: audit.giac.com
Address: 200.200.200.254
```

The query audit.giac.com responds with 200.200.200.254, which the DNS administrator set up on Wednesday. We are now assured that hosts on the internet can not do zone transfers but that GIAC's secondary DNS server can get the zone transfers it requires.

The next rule to test is NTP, port 123 protocol UDP. This can be tested with a UDP scan of clock.psu.edu at port 123. The full command is "nmap –sU –p123 clock.psu.edu". (Note: Penn State requires you contact them for access for NTP synchronization if you are not on their network. The port is blocked for everyone else and shows as closed.) Other UDP scans can be performed at this time. Running nmap against ports 22 and 137-139 on 10.202.0.31 and 5432 to 10.202.0.20, parameters "-sU –p22,137-139,5432 10.202.0.31 10.202.0.20", should show the ports as open. The policy provides four open ports on the first IP and only port 5432 on the second. A follow-up scan using TCP instead of UDP

© SANS Institute 2000 - 2005 Author retains full rights.

should also show the ports as open. Additionally, attempting to browse file shares or make database connections from the appropriate machines will serve to show that the protocol itself works. However, there is a caveat.

All of the rules listed only allow access *from* certain IP's on the service network to the static passthroughs to the internal network. When the tests are run with the laptop's IP of 10.202.0.30, they should all fail. Re-running the tests with the laptop's IP as that of the specified service network machine, such as 10.202.0.5 and 10.202.0.10 when testing port 22/ip to 10.202.0.31, and with that server unplugged from the network should succeed. Unfortunately, the servers must be unplugged for the auditing to proceed, but that is why the audit is performed at this time. The servers should only be unplugged for a few minutes at a time, drastically reducing downtime.

The tests will have to be run multiple times as the laptop's IP is changed. An example of the first UDP test is shown. Note that since the tests are run when the laptop has a source IP of 10.202.0.5, only ssh and PostgreSQL are shown as open; the NetBIOS ports are closed. When the test is run from 10.202.0.10, the NetBIOS ports are shown as open. This is the proof the auditors need to know that the allow *and* deny statements are both working.

```
[root@lankhmar /root]# nmap -sU –p22,137-139,5432 10.202.0.31 10.202.0.20

Starting nmap V. 2.54BETA22 ( www.insecure.org/nmap/ )
Interesting ports on 10.202.0.31:
(The 4 ports scanned but not shown below are in state: closed)
Port       State       Service
22/udp     open        ssh

Interesting ports on 10.202.0.20:
(The 4 ports scanned but not shown below are in state: closed)
5432/udp   open        PostgreSQL

Nmap run completed -- 2 IP address (2 host up) scanned in 6 seconds
```

By running each policy test twice, once from the laptop's IP of 10.202.0.30 and once from the allowed IP addresses for each rule (as above), the auditor proves that the allow rule for the protocol works at the same time he proves the deny other packets rule works. To provide an extra reassurance, the large nmap scans can be performed as well, such as using the web server's IP and running "nmap –p1-5000 10.202.0.31" to ensure that only ports 22, 137, 138, and 139 are open. The output will be very similar to the output shown in the sections above.

The only service network rules left to check, 80/tcp from two web servers to all, must be tested on the other networks.

Both laptops get plugged into the metaframe network next using IP's 10.201.0.3 and 10.201.0.4. The rules for ports 1494/ip, 1605/udp and 81/tcp all work because the auditor was able to connect to metaframe and browse applications when connected to the outside network. Unlike the service network, most of our tests can be done without changing the IP address. The ACL's permit anything on the metaframe network outbound access rather than providing it per IP address. As there is nothing else on the physical network, this is not a problem. If someone were to sneak a machine inside the network and hook it up to the metaframe network, the last two ACL's ensure that the most the person could access would be the few ports specifically allowed by other ACL's. This is not to treat physical security as a non-issue but because any infiltrator could simply unplug the metaframe server and hijack its IP address to get full access. On top of all this, all that is guaranteed is outbound access rather than inbound, making it a mostly useless threat.

Like the service network, most of the same probes can be used in this portion of the audit. Using the same command as in the previous section, the connection to the timeserver can be checked. Syslog ports need to be checked in nmap using the parameters "–sU –p514 10.202.0.8". Running nslookup from either laptop or from metaframe should allow it to connect to 10.202.0.2 and query for any site. The windows laptop and Metaframe should both be able to browse the internal NT servers with the UNC names "\\10.201.0.22" and \\10.201.0.23. TCP and UDP scans can be run to verify the operation of the NetBIOS ports (options "[-sU] –p137-139 10.201.0.22-23"). Metaframe's NetWare client should allow it

© SANS Institute 2000 - 2005, Author retains full rights.

© SANS Institute 2000 - 2005 Author retains full rights.

to browse the Novell server with the UNC name \\10.201.0.30. Nmap can verify the operation of the NCP protocol as well with the options "[-sU] –p524 10.201.0.30". A web browser on any of the three machines should allow it to browse to 10.201.0.29, the internal web server. The Linux laptop will then run "ssh 10.201.0.20" to connect to the internal Unix server; if it gets a login screen and the auditor can log in then the service is working fine. These two services can be verified by nmap with options "-p80 10.201.0.29" and "[-sU] –p22 10.201.0.20", respectively.

Lastly, Metaframe should be able to connect to any machine on the internet on any supported services (i.e. only services that work through NAT). The laptops, however, should not be able to. To ensure that no other services can be reached on the internal network, nmap can be used to probe one or more of the static IP's on ports they should not be able to reach, such as port 25 on 10.201.0.30 or 53 on 10.201.0.20. For example, the command "nmap –p1-1000 10.201.0.20" should only show tcp port 22 as open; the other 999 should be closed.

© SANS Institute 2000 - 2005, Author retains full rights.

```
[root@lankhmar /root]# nmap –p1-1000 10.201.0.20

Starting nmap V. 2.54BETA22 ( www.insecure.org/nmap/ )
Interesting ports on 10.201.0.20:
(The 999 ports scanned but not shown below are in state: filtered)
Port       State       Service
22/tcp     open        ssh

Nmap run completed -- 1 IP address (1 host up) scanned in 60 seconds
```

The laptops will now be moved to the inside network where they will have a variety of IP addresses. There are a number of tests that can be run from the servers rather than the laptops to avoid having to change IP addresses constantly. Other tests have been confirmed by testing on the other interfaces. The rules for port 22/ip from Unix to Metaframe, port 22/ip from the web dev server to the web server and DB server, NetBIOS from the web dev server to the web server, NetBIOS browsing on the two NT machines to Metaframe, file sharing on 524/ip from NetWare to Metaframe, and PostgreSQL connections on 5432/ip from the internal web server to the DB server have all been tested.

The first test is to force the Novell server to sync with clock.psu.edu. The laptop should not be able to sync with clock.psu.edu until its IP is changed to 10.200.20.10. The Novell server should also be able to reach the smtp server at 10.202.0.3. GroupWise will be in constant contact with the SMTP server and it will be fairly obvious if mail is not going through. The only test is to see if machines on other IP's are blocked from going to the SMTP server. Setting a laptop to 10.200.0.200 and running "telnet 10.202.0.3 25" should not result in a connection.

The internal DNS server should be able to do queries on the internet or even on GIAC's own DNS server. A laptop running nslookup will change the server to 10.200.20.7. Queries such as www.giac.com and www.sans.org should return valid results. The laptops should not be able to set their server to anything on the internet, as only 10.200.20.7, the admin stations, and the proxy server receive full access to the internet. If the laptops try and change server to ns1.getwireless.net, they should get an error instead of a connection.

The administrator stations are the next test. The machines should have full access to the internet and telnet access to the PIX. The first auditor has been able to connect to the PIX to look at the ACL's, so the telnet rule is obviously working. If the laptops are still set to 10.200.0.200, they should not be able to telnet to the PIX. Likewise, they should not be able to access the internet, even with the correct DNS server specified. Changing their IP to 10.200.2.200, however, should enable both telnet and internet access. The proxy server, at 10.200.20.5, should also have internet access. Running a web browser on the proxy, or changing a laptop to 10.200.20.5 and unplugged the proxy from the network, will let the auditors know if the rules are working properly.

Before testing the deny rule, the auditors must make sure the external web servers, web access at 10.202.0.10 and webmail at 10.202.0.7, work properly. By setting the browsers on the laptops to use the proxy server, the auditors should be able to access www.giac.com and webmail.giac.com. The deny rule can be tested by trying to access other services that aren't specified, using nmap to probe. From the Linux laptop, with the IP set to 10.200.0.200 again, the command "nmap –p1-1000 10.202.0.10,10.201.0.2,199.199.199.12" should show no open ports on any of the three hosts. The three IP's test all three other interfaces.

The admins will also do some additional tests, of their choice, from different desktops to prove that the denial rules work for other ports. A total of five tests must be run. For instance, the Microsoft Terminal Services client can be used again to try and connect to 10.202.0.7 from a desktop. The connection will fail and a log entry will be added:

```
Feb 20 20:19:19 pix %PIX-4-106023: Deny tcp src outside:10.200.0.14/3987 dst service:10.202.0.7/3389 by
access-group "acl_in"
```

At this point, all of the policies will have been tested. The last step for the auditors is to check the firewall logs for any discrepancy with what was observed during the audit and to look at the numbers of matches on each access list with "show access-list <acl_name>". These numbers should be written down for later perusal.

© SANS Institute 2000 - 2005 Author retains full rights.

© SANS Institute 2000 - 2005 Author retains full rights.

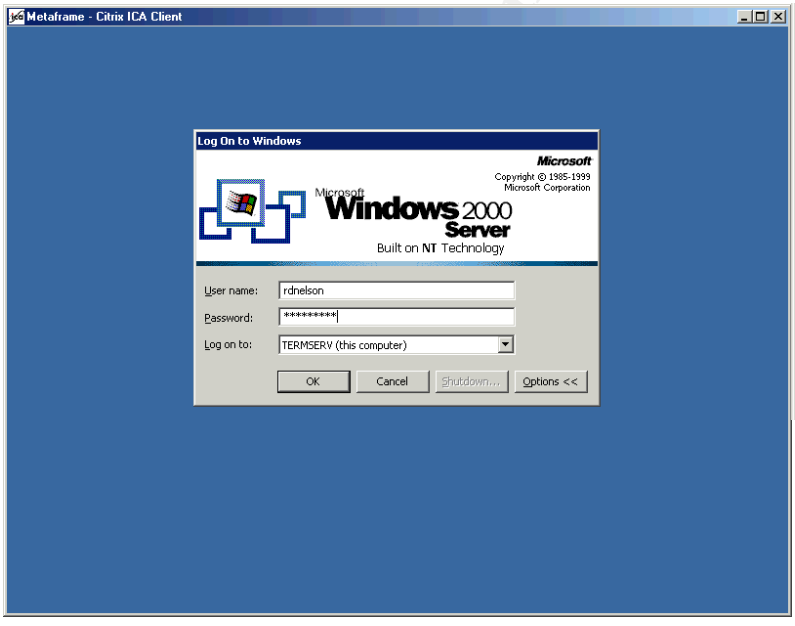
The auditors can write down their results, take care of any serious existing problems, and then go home, taking care of any remaining issues Monday morning. The whole audit, with an efficient team, should take less than 2 hours, barring any problems they may run into.

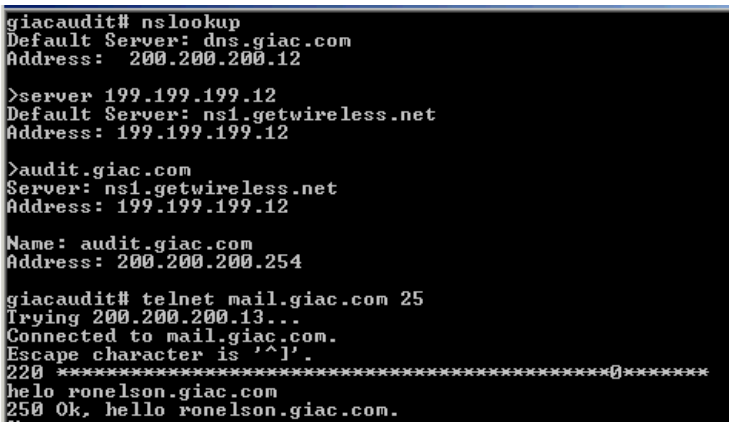
Results of the audit:

However carefully the setup has been planned and maintained, the auditors should expect to find at least one or two problems with the policies or the access lists effectiveness at enforcing the policies. The initial audit, especially, will probably find the most glaring problems that looked fine in theory but didn't hold up to use. The results of the first audit will be discussed below. The following chart summarizes the tests that passed or failed as well as oddities the auditors observed:

Test	Pass/Fail?	Explanation
Access list acl_out	Pass	Results were as expected
Access list acl_service	Pass	Results were as expected; Windows Update doesn't work
Access list acl_metaframe	Pass	Windows update doesn't work; DNS name problems first crop up here
Access list acl_in	Pass	Windows update works with proxy server; DNS name problems again
Windows Update	Fail	Service and Metaframe network are blocked from Windows Update
Server accessed by DNS name fails	Fail	Due to the way PIX handles translations, inside packets that pass through the PIX to the outside and then attempt to pass through to the inside are dropped

The initial portions of the audit are perfectly fine. Since the network is designed primarily with the customer in mind, the attention given the outside setup and policies shows in the accuracy of the access lists. There are no problems in accessing Metaframe, the NFuse web page, www.giac.com, webmail.giac.com, or the DNS and SMTP services of GIAC. Figure 3.1 shows a few screenshots of the individual tests. The nmap portscans used to determine if the denial rule





© SANS Institute 2000 - 2005

Author retains full rights.

works are flawless.

Figure 3.1 – A metaframe client, nslookup results, and the SMTP header

The audit of the service network is flawless as well. From the bogus DNS record for audit.giac.com, to time synchronization, to file browsing and ssh, the tests perform as expected. A drawback is found, however, in that outbound access is severely restricted. One of the auditors accidentally started Windows Update, which could not connect to the update servers. That is what the policy states, but it will make updating the servers much more difficult and time-consuming.

The metaframe network's audit falls a little short. File browsing, ssh, logging, time synchronization, and DNS all work fine. Windows Update is not a problem because outbound web access is allowed. However, accessing certain servers by their DNS names doesn't work. Trying to access the DNS server as dns.giac.com fails! The name resolves correctly to 200.200.200.12, so the auditors look at the problem. This is where tcpdump comes in handy. By running tcpdump on the Linux laptop on the service network, the auditors looked for an incoming connection, expecting to see the acknowledge packets get blocked. The command was "tcpdump –i eth0 –nn port 53", -i specifies the interface to use and –nn tells it not to look up DNS names or port protocols. No packets, incoming or outgoing, were seen, however. Removing the "port 53" filter didn't show any sort of connection between the two machines on any port. Using the internal IP address, 10.202.0.2, did work.

The auditors moved the Linux laptop to the outside interface and ran tcpdump again. This time, packets were seen outgoing, as expect. See below. No return packets were seen. The auditors didn't know where else to look, so started checking cisco.com's search engine for "connection to host by name failing + PIX". After a while, they come up with the **alias** command for the PIX (http://www.cisco.com/univercd/cc/td/doc/product/iaabu/pix/pix_61/cmd_ref/a.htm#65183).

```
[root@lankhmar /root]# tcpdump -nn -i eth0
tcpdump: listening on eth1
20:57:28.499344 200.200.200.10 > 200.200.200.12: icmp: echo request
20:57:29.490083 200.200.200.10 > 200.200.200.12: icmp: echo request
20:57:30.490083 200.200.200.10 > 200.200.200.12: icmp: echo request
20:57:31.490086 200.200.200.10 > 200.200.200.12: icmp: echo request
```

The description of the problem is sparse, but the two auditors figured out what happened. When the internal machine (10.201.0.2) tried to access dns.giac.com (200.200.200.12), the packet went from the metaframe network to the outside network. On the outside network, the packet looks like it has a source of 200.200.200.10, the static address, with a destination of 200.200.200.12. The PIX treats it as two machines on the same subnet and ignores it. Of course, the PIX is supposed to respond for 200.200.200.12 and pass the packet to the DNS server, but it doesn't. Cisco offers no reason *why* the PIX does this, but it does offer a workaround. Figure 3.2 shows the path the packet tries to take to the DNS server.

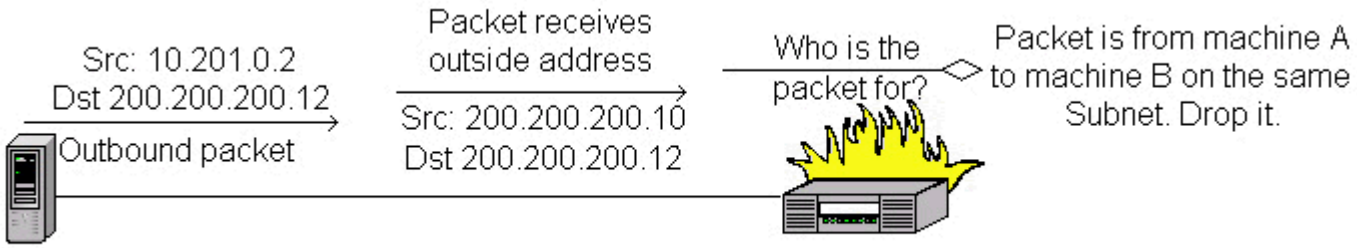


Figure 3.2 – What happens to a packet traveling to a static IP on the same subnet

The alias command was designed to fix this. The alias command takes four arguments: an interface the internal address, the external address, and a netmask. Any DNS query that pass *through* the PIX will be checked. If the query includes an external address specified in an alias statement and goes through the specified interface, the IP will be changed with the internal address. In this case, the request for dns.giac.com will return 200.200.200.12. The request will be checked on

© SANS Institute 2000 - 2005

Author retains full rights.

the PIX and modified. The answer that Metaframe sees will say that dns.giac.com is reachable at 10.202.0.2. Now, packets will be able to travel between the two machines. The command to run on the pix is:

```
alias (metaframe) 10.202.0.2 200.200.200.12 255.255.255.255
```

There are two drawbacks to this. First, a list of alias commands must now be maintained. The list should be relatively short, no more than a dozen, a minor aggravation. Second, the alias command only works on traffic that passes through an interface. If the SMTP server were to attempt to communicate with the webserver by using "www.giac.com", no alias command would be processed and an impossible connection would start to try and connect with 200.200.200.20. The fix for this is definitely less graceful, a **hosts** file on every server in the service network. The auditors, however, decide to only implement the alias command. None of the machines on the server network have any real need to talk to each other via the DNS name at this time. The auditors take note, in case a future situation introduces that need.

The internal audit runs into similar problems. Audits of NetBIOS, NetWare file sharing, DNS, time synchronization, ssh, PostgreSQL, and web connections work fine. Admin stations and the proxy server are unrestricted. The test for denial works similarly and the policies are all verified to be correct. Windows Update works on machines that use the proxy server, so proxy settings are enabled on the servers. The same DNS name problem crops up almost immediately.

While none of the employees have need to connect to the external webserver for the web access it provides, the administrators occasionally have need of it. When the auditors test access to www.giac.com and webmail.giac.com, they can't access them because they are pointed to the outside IP addresses. Three more aliases need to be created, including dns.giac.com. The complete list of aliases is:

```
alias (metaframe) 10.202.0.2 200.200.200.12 255.255.255.255
alias (inside) 10.202.0.2 200.200.200.12 255.255.255.255
alias (inside) 10.202.0.7 200.200.200.17 255.255.255.255
alias (inside) 10.202.0.10 200.200.200.20 255.255.255.255
```

The entry for dns.giac.com needs created again specifically for the inside network. Because the DNS server is on the service network, allowing the alias command to take over for all interfaces would then provide the outside world with the address 10.202.0.2. Quite obviously, not many people would be able to get to the DNS server and it would cause numerous connectivity issues.

Fixing Windows Update is a minor issue. It would be simple to enable port 80 access to the internet for the machines on the service network with the additional rule "access-list acl_service permit tcp any any eq 80". However, Microsoft is preparing products to allow companies to host their own internal Windows Update repository. GIAC will wait to evaluate the local repository feature before enabling direct Windows Update. As mentioned before, it is almost always more difficult to tighten security later than it is to loosen it. There is also a risk of a user at the desktop of one of the servers visiting a harmful web page – a small risk, perhaps, but with greater penalties that not having direct access to Windows Update.

The audit is complete with only a few problems; above average for an entirely new network. The auditors notice nothing untoward or conflicting in the syslog entries and go home happy, only 30 minutes late. The access list numbers can be looked at on Monday.

The access lists show how many times each rule has been matched. The PIX traverses the entire ruleset until it finds a match. A denied packet must be checked against every rule before it fails, part of the reason the access lists are kept as short as possible. However, permitted packets may still need to be checked against many rules. The network admins take a look at the access lists and the numbers of matches on Monday morning. Examining the acl_out list, they are provided with this data:

```
access-list acl_out permit tcp any host 200.200.200.10 eq 1494 (4732)
access-list acl_out permit udp any host 200.200.200.10 eq 1494 (4894)
access-list acl_out permit udp any host 200.200.200.10 eq 1605 (312)
```

© SANS Institute 2000 - 2005

Author retains full rights.

```
access-list acl_out permit tcp any host 200.200.200.10 eq 81 (125)
access-list acl_out permit tcp any host 200.200.200.13 eq smtp (48293)
access-list acl_out permit tcp host 199.199.199.12 host 200.200.200.12 eq domain (2438)
access-list acl_out permit udp any host 200.200.200.12 eq domain (938132)
access-list acl_out permit udp host 128.118.25.3 any eq 123 (683923)
access-list acl_out permit tcp any host 200.200.200.20 eq www (1948205)
access-list acl_out permit tcp any host 200.200.200.17 eq www (50382)
access-list acl_out deny ip any any (8392018)
```

By far, the majority of matches are against the deny rule. However, that rule must be last or valid traffic would be denied. The next most popular rule is line 9, nearly 2 million packets to www.giac.com. Third popular is line 7, almost 1 million packets to the DNS server. Because of the ordering of the rules, 2 million packets had to be checked against 9 rules and 1 million against 7 rules. By moving these two rules to the top, nearly 22 million rules checks can be eliminated. Sorting the remaining rules by popularity can provide other improvements. The resulting order is the best optimization for the acl_out rule:

```
access-list acl_out permit tcp any host 200.200.200.20 eq www
access-list acl_out permit udp any host 200.200.200.12 eq domain
access-list acl_out permit udp host 128.118.25.3 any eq 123
access-list acl_out permit tcp any host 200.200.200.13 eq smtp
access-list acl_out permit tcp any host 200.200.200.17 eq www
access-list acl_out permit udp any host 200.200.200.10 eq 1494
access-list acl_out permit tcp any host 200.200.200.10 eq 1494
access-list acl_out permit tcp host 199.199.199.12 host 200.200.200.12 eq domain
access-list acl_out permit udp any host 200.200.200.10 eq 1605
access-list acl_out permit tcp any host 200.200.200.10 eq 81
access-list acl_out deny ip any any
```

The other access lists will benefit similarly from this. A suggested re-ordering is offered in the appendix under “Suggested Re-Ordering of PIX ACL’s” based on expected numbers of matches, although only an audit can show the most effective order. An important side effect of the audits is the organization of the access lists based on popularity. Usage of certain services is bound to fluctuate over time and a constant re-evaluation of the ordering can only be beneficial. As shown by the elimination of over 17% of required rule checks, including this step in the 3-month audits as well can help GIAC prolong the life of the firewall significantly.

Regardless of the turnout, the audit is a success. The auditors have verified that most services behaved as they were supposed to. Those that were not have been corrected. The auditors also found out that they need to enable a few alias commands to properly serve internal users. The network has shown its ability to be modified quickly and to be optimized according to the users’ priorities. GIAC is now in possession of a PIX firewall and a set of access lists that upholds their security without compromising business needs.

The firewall’s effectiveness at enforcing the policies is 100%. The perimeter defense of GIAC Enterprises is very thorough. By cleaning up the ordering of the rulesets the performance is kept high. Adding the alias commands allows consistent access across the four different networks. The audit itself shows where the policy itself has weak spots, such as denying Windows Update, and gives the network administrators a handle on how the policy matches the users’ needs.

By using large ranges of port addresses during nmap scanning, the auditors can be reasonably certain that not only is the firewall providing the expected functionality but also proving that the security policy is working by blocking probes and attacks against unused services and protocols.

There were no outright failures, but it is important for the auditors to document the problems with using DNS names on the different networks and with accessing Windows Update. The resolution to using DNS names, the alias command, is only a partial fix, as the service network machines will still get the external IP on a DNS lookup. The auditors have decided not to use a host file on every machine to fix this, but it may cause problems in the future. The alias command

© SANS Institute 2000 - 2005 Author retains full rights.

does fix the problem for computers on the service and internal network as well as adding to the administrative overhead of the firewall. Changes to the DNS information for giac.org will need to be checked against the firewall so that they are kept in sync.

While there are plenty of alternative network architectures that will handle the problems that GIAC encountered with the audit, they all bring their own problems to the table. Perhaps the simplest change that can be made is modifying the policy for web access on the servers. Providing open web access from the servers would allow the administrators to do more administrative work from the servers themselves. Currently, if the administrator wants to investigate a problem with the OS or an application, he or she must move to their own machine to look at the vendor's site or download a patch. Opening the policy up for web access would allow patches to be downloaded from the server as well as running Windows Update. This would be a worthwhile change regardless of Microsoft's ability to provide local Windows Update repositories.
Not much can fix the DNS problem. External hosting would provide the same need for alias statements. Using internal DNS servers with a modified version of the database would increase the administrative overhead with no real benefit. The only change that could entirely rid the network of this problem would be to replace the PIX with another firewall that allows packets to be translated to the outside interface *and* to come back through the PIX. Making this change, however, would require a complete overhaul of the firewall policy and implementation, not to mention an entirely new audit procedure and purchasing rounds. It is not an entirely feasible change at this time, due to the recent acquisition of the PIX, but is a topic that can be revisited when the time comes to upgrade or replace the PIX in a few years.

Summary:

With a successful 6-month audit to start out, GIAC's network is enforcing the current policies. As requirements change and policies get updated, it is always important the effectiveness of the policies and the devices be audited. Not only will audits protect the network from malicious attack but they also help provide a clear idea of what business needs are or are not being met.

© SANS Institute 2000 - 2005 Author retains full rights.

Assignment 4 – Design Under Fire (25 points) The purpose of this exercise is to help you think about threats to your network and therefore develop a more robust design. Keep in mind that the next certification group will be attacking your architecture!

Select a network design from any previously posted GCFW practical (http://www.giac.org/GCFW.php) and paste the graphic into your submission. Be certain to list the URL of the practical you are using. Research and design two of the following three types of attacks against the architecture:

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

Your attack information should be detailed - include the specifics of how the attack would be carried out. Do not simply say "I would exploit the vulnerability described in Vendor Security Bulletin XXX". What commands would you use to carry out the attack? Are exploit tools or scripts available on the Internet? What additional steps would you need to take prior to conducting the attack (reconnaissance, determining internal network layout, determining valid account name.)? Would any of your methods be noticed (log files, IDS.)? What "stealth" techniques could you employ to avoid detection? What countermeasures would help prevent your attack from succeeding?

If it is possible to carry out the attack on a test system, include screen shots, log files, etc. as appropriate to illustrate your methods.

In designing your attacks, keep the following in mind:

- *The attack should be realistic. The purpose of this exercise is for the student to clearly demonstrate that they understand that firewall and perimeter systems are not magic "silver bullets" immune to all attacks.*
- *The attack should be reasonable. The firewall does not necessarily have to be impenetrable (perfectly configured with all of the up-to-the-minute patches installed). However, you should not assume that it is an unpatched, out-of-the-box firewall installed on an unpatched out-of-the-box OS. (Remember, you designed GIAC Enterprises' firewall; would you install a system like that?)*
- *You must supply documentation (e.g., a URL to the security bulletin, bugtraq archive, or exploit code used) for any vulnerability you use in your attack.*
- *The attack does not necessarily have to succeed (though a successful attack is often the more interesting approach). If, given the perimeter and network configuration you have described above, the attack would fail, you can describe this result as well.*

© SANS Institute 2000 - 2005, Author retains full rights.

Section 4 Layout

1. Defining the need ("The purpose of this exercise is to help you think about threats to your network and therefore develop a more robust design")

    - Discuss Daniel Martin's design ("Select a network design from any previously posted GCFW practical and paste the graphic into your submission. Be certain to list the URL of the practical you are using")

2. The plan ("Research and design two of the following three types of attack against the architecture…")

    - Use a Denial of Service attack to distract GIAC Fortunes IT administrators ("A denial of service attack. Subject the design to a theoretical attack from 50 compromised cable modem/DSL systems")
    - Compromise an internal machine via social engineering ("An attack plan to compromise an internal system through the perimeter system")
    - A sympathy email will be mass-mailed to specific GIAC Fortunes employees the attacker has previous contact with to deliver the virii ("What additional steps would you need to take prior to conducting the attack")
    - An attacker, "Boris", will try these attacks simultaneously in order to distract administrators from an attempt on their Oracle database ("Select a target, explain your reasons for choosing that target")
    - Boris will pose as a Russian fortune cookie translator in order to gain the trust of GIAC Fortunes ("What stealth techniques could you employ to avoid detection?")
    - Boris will use SubSeven for DoS and two virii for compromising the internal host ("describe the process to compromise the target")
    - SubSeven zombies will launch a DoS with iis5hack ("You must supply documentation for any vulnerability you use in your attack")

3. The payload ("Your attack information should be detailed – include the specifics of how the attack would be carried out… What commands would you use to carry out the attack? Are exploit tools or scripts available on the internet?")

    - W97M.Pacol.A and W32.Klez.E@mm viruses will be used to compromise the internal machine via email ("What commands would you use to carry out the attack?" and "You must supply documentation for any vulnerability you use in your attack")
    - Include commands that SubSeven will run. ("What commands would you use to carry out the attack?")
    - Discuss how the attacks will be seen. ("Would any of your methods be noticed (log files, IDS.)?")

4. The attack - Describe the details of Boris's actual attack ("include the specifics of how the attack would be carried out" and "The attack does not necessarily have to succeed. If, given the perimeter and network configuration you have describe above, the attack would fail, you can describe this result as well")

5. The results ("If it is possible to carry out the attack on a test system, include screen shots, log files, etc. as appropriate to illustrate your methods.")

    - Discuss problems that allowed Boris to attack and remedies ("What countermeasures would help prevent your attack from succeeding?")

6. Summary

SANS Institute 2000 - 2005, Author retains full rights.

Section 4: Design Under Fire

Defining the need:

As has been repeated often, security is a process, not a product. It is not static and needs updating constantly. Administrators as well as users need to be well trained and educated properly.

To provide an example of this, another network has been analyzed for its shortcomings. It can not simply be assumed that the administrators have not touched the network since it was installed, but it is fairly safe to presume that not all required updates will have been made on every system. Time and staff in most organizations are amongst the most rare resources, often precluding a dedicated security officer who can ensure the integrity of every system. GIAC will, sooner or later, be afflicted with this problem. It should be anticipated that over time, the overall level of security of GIAC's network will start to degrade as more time must be spent on other tasks rather than security updates.

The target network is the GCFW practical of Daniel S. Martin, found at http://www.giac.org/practical/Daniel_Martin_GCFW.doc. Daniel's design was not targeted for any specific reason; in fact, it is good, solid network design. However, multiple attacks against IIS5.0 have been publicized since Daniel's design was submitted and has had 10 months to slowly degrade. Daniel's network diagram is included below in Figure 4.1.

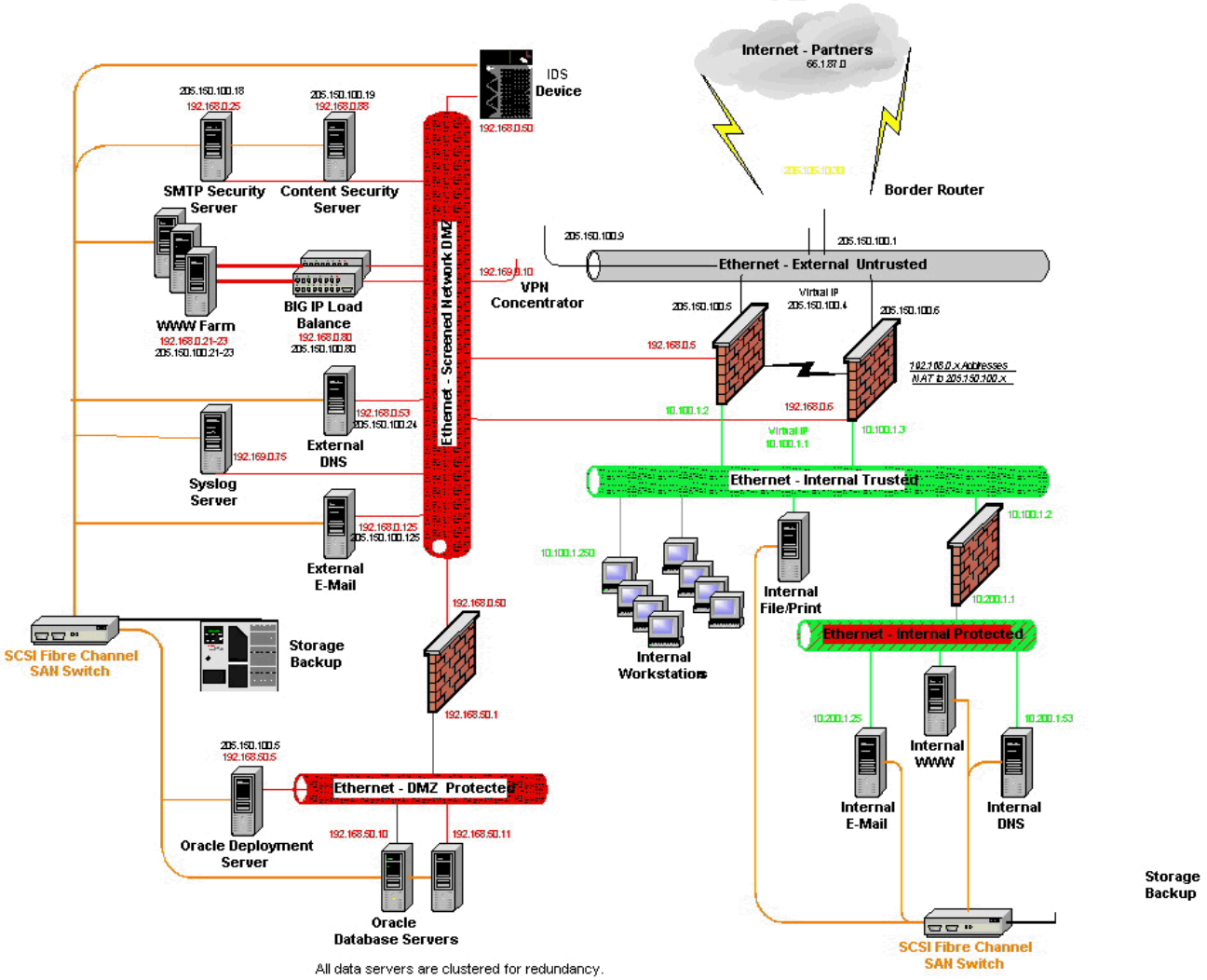


Figure 4.1 – Daniel Martin's Network Diagram

There are three primary methods to attack a network. First, an attacker can try and exploit the border router or firewall, be it a "root" exploit or some form of a denial exploit. Second, an internal machine may be compromised to allow remote access, capture passwords, or forward sensitive data to some outside repository. Third, a Denial of Service (DoS) attack – that is, an attack that prevents a service from being used, usually attacking the ISP to Company link – can be launched to severely hamper the operations of a company.

The plan:

A sample attack will be shown using a DoS attack and internal compromised machines simultaneously. An attacker coordinating such an attack can hope that the limited resources of available staff will allow one attack or the other to persist longer than could be expected if it were the only attack. A widespread attack of desktop systems, followed by an attack on the web farm, could easily provide such an opportunity.

A few assumptions must be made about the attack. To provide the degree of coordination needed for the attack, it must be assumed that the attacker is fairly intelligent, or at least proficient at his "business". To prepare for the denial of service, the attacker will have already compromised at least 50 consumer machines on some form of broadband connection, such as cable modems or DSL. The tool used to compromise the machines is Subseven, allowing almost complete control over Windows 98 machines. A complete list of features in Subseven can be found at http://www.europe.f-secure.com/v-descs/subseven.shtml. The primary features Boris wants are the ability to upload files and to enter a manual command. Subseven zombie machines can be controlled via IRC, which is how Boris can signal them to attack.

The attacker has also compromised a host in Russia, example.ru, and set up a mail account there with the name Boris Bruschev. He has been in contact with the sales team to open an account as a supplier and translator. After providing some sample fortunes the attacker collected from taglines on a Russian mailing list, Mr. Martin's network has been modified to allow access from the compromised host for the false supplier. From this, he has garnered a few items, including a method to access the internal Oracle server, some email addresses of the sales people (Joe Smith, jsmith@giac-fortunes.com), accounting (Wanda Eldo, weldo@giac-fortunes.com), and billing (Sally Johnson, sjohnson@giac-fortunes.com). Calling from a European produced cell-phone, purchased under the auspices that Russian cell-phones are not as great outside of Russia itself, Boris has talked to Wanda and Sally frequently. Boris has made a huge deal about the September 11th attacks, apologizing profusely and offering sympathy, in the hope of gaining some trust with the two.

Boris lives in the Ukraine, so it is not out of the realm of possibility that he would be able to handle all the aspects of this problem. He knows a bit of Russian and has friends who can help with the rest. He buys a cell phone on a trip west, to Germany, and sets up a bank account in Sweden. Boris is not new to the business of cracking and has more than a few years experience under his belt. It seems that Boris's intent is to attack GIAC Fortunes and make some money at the same time.

By using a false identity, attacking from a country with weak law enforcement, and purchasing equipment from other countries, Boris creates a long trail between the attack and himself. Swiss bank accounts are often set up with anonymous locks are "passes" so that the account holder can not be tracked or even verified. This is about as stealthy as an attacker can be without living underground or paying off high-level law enforcement. The cost of tracking Boris alone is likely to outweigh the damage he can do to GIAC Fortunes. On top of all this, when Boris makes his attack, he can continue to go about his daily business as normal without any significant change in his lifestyle. The protection afforded to him by his anonymity and the difficulty of tracking him provide nearly perfect insurance.

The attack can now be fleshed out. Boris will include Wanda and Sally in a mass-mailing with the subject "Help September 11th Widows!" The message will include many forwards to make it look like a popular message. The text will be in English, but appearing to have been written by a Russian with rudimentary English skills. It will announce that a certain Russian ISP will donate $.10USD for every person who clicks on the attachment, then ask for each recipient to forward it to 10 more. The exact text of the message, minus all the forwards, is as follows:

© SANS Institute 2000 - 2005 Author retains full rights.

To all fellow Russians!

America have suffered much since 11 September! As great friends of Mother Russia, America deserve our respects! Internet providers ROL, Russia On-Line, want to help! For every persons who read this message and click on attachement attached, ROL will donate 10 cents to 11 September Fund for all Widows of attack!

This great way to help great Russian Friend. Please pass to all your friend and ask them to do the same! Help our America Friend!

D. Cheyvic,
ROL

Boris attaches a potent new virus (or three!) with hopes that GIAC Fortunes has not employed Anti-Virus updates to counter it. A relatively new virus that requires a lot of cleanup on the desktop fits perfectly. It doesn't need to be the most powerful of viruses, but mildly damaging and difficult enough for removal to require some of GIAC Fortunes' IT staff to visit the desktops to remove it. The window of opportunity is small, only about 24 hours, but very useful. It is likely that Boris would *not* be the only person forwarding the virus to GIAC Fortunes employees, preventing any undo suspicions about Boris's intentions…yet.

Boris may also attempt to wait a while before trying this attack. Newer Anti-Virus programs have stopped providing free updates an only allow free updates for 12 months. After 12 months, updates complete without an error but don't actually update the definition files. Mr. Martin's paper is dated March 28th, 2001. Attacking in early April through late May of 2002 with a new virus may be more effective if GIAC Fortunes has allowed the definitions to run out. Even without waiting for AV updates to run out, which GIAC Fortunes may renew, Boris has some opportunity to attack while other companies are overloading the servers providing the AV updates. The Melissa virus, and others, was so effective because the servers providing updated files to protect against Melissa were unreachable. In some cases, companies weren't able to get updates or fixes until three days after the virus hit the news.

Boris performs his mass-mailing the evening before his denial of service attack (Mr. Martin does not specify what time zone his company is in). Sally and Wanda should have received their mails by the morning and, with any luck on Boris's part, left their computers on. If Boris sends an auto-forwarding virus and the user left their mail client on, it could have propagated by the time the day shift begins. When Wanda and Sally start their day, they will probably pass it on to other employees who may run the virus as well. Boris now has at least two internal machines that have been comprised, possibly many more.

At 9am, Boris will launch the second part of his attack. His 50+ compromised home machines are focused on GIAC Fortunes. The web servers run IIS5.0, known to have a number of vulnerabilities, including some Denial of Service attacks. Boris doesn't know for sure, but suspects that there may be a web farm. By dividing his compromised machines into 5 groups of 10, he hopes that the load balancer will pass at least one client to each web server. Each group will launch 30 seconds apart with a delay of 210 seconds, providing 10 machines hitting the web servers every 30 seconds, continuously. There is a fairly recent IIS vulnerability that Boris will exploit, the Microsoft IIS 5.0 .printer ISAPI Extension Buffer Overflow, detailed below, with a program called "iis5hack.exe".

To re-cap, Boris's attack looks like this:

- 10pm the night before, send mass-mailing to GIAC Fortunes employees
- 9am, the employees have hopefully activated the virus and GIAC Fortune's administrator's are fixing the desktops
- 9am-12am, 50+ individual machines attack the web servers with a denial of service attack on Boris's command.

At this point, Boris will have caused massive confusion at GIAC Fortunes. It is not unreasonable to believe that most of the technical staff is dealing with either desktop issues or with web servers that are unreachable. Worse, Boris has an account on GIAC Fortunes' database and plenty of time in which no one is watching him. There have been many publicized break-ins with both Oracle 9i and Sun Solaris 7, the OS and database that Boris gets access to through the VPN. It will not take long for Boris to gain access to the privileged portions of the database and view customer lists,

© SANS Institute 2000 - 2005

Author retains full rights.

account information, and potentially tamper with the data. Depending on how long it is before the administrators get back to watching all aspects of their network, Boris could slightly modify the database, erase his tracks, and leave without the administrators being any wiser.

If Boris's attack goes well, GIAC Fortunes will take a licking and lose a significant amount of money and time in repairing the damage done. That's Boris's plan, anyway. The details will determine if it works or not.

The payload:

The virus payload is the first choice Boris has to make. Boris is looking for a virus that is new, requires a long time to repair, and spreads on its own. Visiting Symantec's "Symantec Security Response" page (http://securityresponse.symantec.com), Boris takes a look at the top 10 list. Two viruses on the top ten [Note: Using the top 10 list from 1/27/02 for this example] are particularly appealing. W97M.Pacol.A, a Win32 Macro and Trojan Horse, (http://securityresponse.symantec.com/avcenter/venc/data/w97m.pacol.a.html) has a more noticeable payload that's easy to remove. However, being noticeable is useful for the purposes of distraction. The virus lowers security levels in Microsoft Word 2000 and XP and overwrites .txt, .doc, .wri, and .pdf files. Pretty noticeable to people in accounting and billing!

However, there are other viruses, and it seems that the W32.Klez.E@mm Win32 worm/virus (http://securityresponse.symantec.com/avcenter/venc/data/w32.klez.e@mm.html) might be better for Boris to use. The virus disables command anti-virus products; deletes checksum database files; copies itself to local, mapped, and network drives; uses a double file name extension (*.txt.exe and *.txt.rar); and searches a variety of local databases for email addresses and emails them to an address. The removal instructions are fairly lengthy and complicated, involving administrators using regedit.

While it is not a trivial task to get ones hands on a copy of the virus, particularly when it is new, it is assumed that Boris will be able to get a hold of it in some form. Boris will take both viruses, put them in an exe file, and package it so that both viruses will be run when then user clicks on their email attachment. There's no reason for Boris to presume that one virus alone will be noticeable, so he uses both to ensure maximum chaos at the site. By using Winzip's "Zip2Exe" feature, Boris can package two virus files in one zip file and run both files when unzipping is complete. The user will see an attachment "ROnline.exe" at the end of the email. When they double-click on it, it will extract "ROL.exe" and "ROL2.exe" to the hard drive and run ROL.exe. This file will contain W98M.Pacol.A and also run ROL2.exe, containing W32.Klez.E@mm. The email payload is now complete. The next step is to provide the denial of service attack.

The most recent DoS that is unlikely to have been patched is the Microsoft IIS 5.0 .printer ISAPI Extension Buffer Overflow. As the SecurityFocus discussion (http://www.securityfocus.com/bid/2674) notes:

> "Windows 2000 Internet printing ISAPI extension contains msw3prt.dll which handles user requests. Due to an unchecked buffer in msw3prt.dll a maliciously crafted HTTP .printer request containing approx. 420 bytes in the 'Host:' field will allow the execution of arbitrary code. Typically a web server would stop responding in a buffer overflow condition; however, once Windows 2000 detects an unresponsive web server it automatically performs a restart. Therefore, the administrator will be unaware of this attack.
>
> "*If Web-based Printing has been configured in group policy, attempts to disable or unmap the affected extension via Internet Services Manager will be overridden by the group policy settings."

Boris is relying on GIAC Fortunes to have not patched the server for this fix or to have left the ISAPI extension enabled. While Windows 2000 does ship with the Internet Printing extensions enabled, it is fair to assume that any administrator worth his or her salary will have disabled it. Given the spread of viruses like Nimda, that rely on old exploits in products that should have been patch months ago, however, it can be seen why attackers like Boris would make the attempt. Constantly restarting the web server is likely to disrupt any persistent sessions that e-commerce apps would use and possibly cause the CPU of the web server to spike.

SecurityFocus offers the program IIS5Hack.exe (click on Exploit in the vulnerability listing) in a ZIP file that includes C

© SANS Institute 2000 - 2005, Author retains full rights.

© SANS Institute 2000 - 2005 Author retains full rights.

and Perl source code that will exploit the vulnerability. Undoubtedly, there are other more malicious programs out there to exploit the same vulnerability. IIS5Hack takes three parameters, the target host, a netcat port and a netcat host. If the target is vulnerable to this attack, IIS5Hack also creates a shell for the user to connect to. Boris is uninterested in this "feature" since his real target is the Oracle database. The zombies will be given a specific command:

```
C:\>iis5hack 205.150.100.80 81 205.150.100.80
```

With 50 machines attacking at once, the web service will be very unusable as it continually restarts. Boris also protects his "investment" by making sure that no one can connect to the hi-jacked webservers by opening the netcat port to the webserver itself.

Boris's attack itself is fairly stealthy as well. While his actions would be seen in a log file – first, the email coming in with a virus attached and then later a denial of service attack – the actions originate from disparate hosts on the internet and are not related by time of attack. Tracing both attacks back to Boris will take more than 6 hours, putting any discovery well after the time of Boris's "disappearance".

The attack:

The mass mailing is made at 10pm on January 27th, 2002. Boris just attaches his virii to the chain letter and forwards it to Sally and Wanda. He bounces it off an open relay of a local ISP so that it can't be used to track directly to him. It arrives in GIAC Fortune's email system around 10:20 and is distributed to the users.

Boris's attempt to give the virus to desktop users is successful. Not only does it rely on the fairly reliable tendency of users to fall for a sap story and run the attachment, but also repackaging it under a different name increases its likelihood of getting past virus filters. It does, and both Sally and Wanda get both viruses. Neither user leaves their computer on the night before, but when the log in on January 28th, 2002, they both find an email chain letter asking them for nothing more than to double-click on a file. They do, and then send it on to several co-workers, who do the same. By 8:30, the IT department has had three phone calls – ironically, from Sally, Wanda, and Joe Smith – from people who cannot find some of their documents on the network. By 9am, five of the IT staff – 3 technicians and two administrators, leaving only the receptionist to answer the phones – are involved in fixing desktops, restoring files, and updating the filters on the email server.

Boris is a little late getting back from the café, so doesn't launch his denial of service attack until 9:10. Fortunately for GIAC Fortunes, they have patched their servers and removed the unused ISAPI association. Boris's denial of service goes unnoticed, having no effect on the web servers and a negligible effect on the speed of the internet connection. Poor Boris.

However, Boris has about 6 hours to work undetected. The IT department spends all day working on desktops. It is only at 4:30, when they go back to the office to check email and take a quick look at the network and the servers that they have any time to look at the database. Boris spent about 3 hours working on the database before giving up. Oracle 9i is, at least to Boris, unbreakable.

The timeline ends up looking something like this:

Time	Action	Result
10pm, January 27th	Chain Letter sent	Distributed to Sally and Wanda's desktops
8am, January 28th	Users log in, forward sympathy letter to co-workers	Virus infects Sally and Wanda's machines and is passed to others
8:30am	IT workers receive phone call about "odd behaviors" from some users	Virus starts to spread through GIAC Fortunes
9am	Five of the IT staff are removing virii from computers	Entire IT staff, minus the receptionist, becomes distracted by the virii

© SANS Institute 2000 - 2005 Author retains full rights.

9:10am	Boris launches iis5hack via 50 compromised hosts running SubSeven	Web servers are attacked by a vulnerability they are patched for. No damage done
9:15am	Boris attacks the Oracle database	Even thought the denial of service attack fails, the IT staff is busy all day long and never notices Boris's frantic activity
12:30pm	After trying for 3+ hours, Boris gives up in his futile attempts to crack the database	The Oracle database is left intact and Boris disappears forever

The results:

GIAC Fortune has suffered some damage, but nowhere near what it could have been. Boris was able to make the IT department waste a whole day doing nothing but fixing what he broke. He also had sold some fortunes and got paid for them. GIAC Fortunes gets to use the fortunes, but they have unfortunately paid the very person attacking them. No permanent damage has been done to the database; this is especially fortunate as no one realized Boris was attacking it until two weeks later, when they realized that Boris didn't really exist and that their last communication with him was a fake chain letter.

A few problems existed that allowed Boris's attack to even begin. The first is a very lax policy in account creation. On top of this, the IT department didn't require much authorization to set up Boris as VPN partner. Anti-virus definitions weren't updated at a regular pace, allowing the virus attacks to distract the IT department for a whole day. The only successes that the IT staff had were their preventative maintenance of the web servers and a secure database.

Most of the problems could have been fixed very simply with constant updates and some attention from management and purchasing toward security. A centralized anti-virus update server, as well as a paid subscription with support from Symantec, would allow the administrators to push updates down to the clients without having to visit them. Since the discovery of the newer virus was 6 days previous, there's a good chance automated updates would have handled the problem. At the least, the older virus, 11 days old, would have been stopped.

The denial of service attack was stopped because of preventative maintenance. Server updates are, and should be, a regularly scheduled part of someone's job. Disabling unused extensions is another good practice. Evidently, the hardening of Oracle was effective since Boris was kept out of the privileged portions.

The only serious problem that can't be remedied by buying a subscription or having the IT staff is that Boris was able to create an account. Joe, Wanda, nor Sally made any verification past a phone number, an account, and a few fortunes that Boris passed off as his own. GIAC Fortunes should implement a much stricter policy for account creation. Requiring a real life address, verifiable by my mail, would provide a good foundation. It could be expanded upon by requiring a copy of a photo ID, landline phone and fax (as opposed to cellular), and a recent picture of the primary contact.

Summary:

Attacks on a network may come from any direction. Boris may have had a run-in with GIAC Fortunes or an employee that left bad feelings or Boris may simply have needed some quick cash. Maybe someone else hired Boris; maybe he was simply bored. Whatever the reason, Boris's small amount of money and effort was ultimately unable to gain him access to his objective, an Oracle database, and probably cost him more than he gained.

GIAC Fortune's policies were responsible both for allowing him illicit access and for preventing him complete access. A lack of detail in the account policy opened the door, but a strict adherence to server-room policies saved the company a lot of time and money.

© SANS Institute 2000 - 2005 Author retains full rights.

© SANS Institute 2000 - 2005 Author retains full rights.

Appendix:

Router Config: Cisco 1605R running IOS v12.2(6)a

```
version 12.2
no service finger
service slave-log
service password-encryption
no service udp-small-servers
no service tcp-small-servers
!
hostname Router
!
enable secret $1$Ohe0$lx2f2EX1Ts7T4l5ivuUE0.
!
no ip http server
no ip bootp server
no ip source-route
!
interface Ethernet0
 ip address 200.200.200.1 255.255.255.0
 ip access-group 20 in
 no ip redirects
 no ip unreachables
 no ip directed-broadcast
 shutdown
!
interface Ethernet1
 ip address 199.199.199.126 255.255.255.64
 ip access-group 20 in
 no ip redirects
 no ip unreachables
 no ip directed-broadcast
 ntp disable
 no fair-queue
!
no ip classless
logging buffered
logging 200.200.200.25
access-list 1 permit host 200.200.200.3
access-list 20 deny   10.0.0.0 0.255.255.255 log
access-list 20 deny   172.16.0.0 0.15.255.255 log
access-list 20 deny   192.168.0.0 0.0.255.255 log
access-list 20 deny   127.0.0.0 0.255.255.255 log
access-list 20 deny   224.0.0.0 31.255.255.255 log
access-list 20 deny   0.0.0.0 0.255.255.255 log
access-list 20 deny   1.0.0.0 0.255.255.255 log
access-list 20 deny   2.0.0.0 0.255.255.255 log
access-list 20 deny   5.0.0.0 0.255.255.255 log
…
access-list 20 deny   219.0.0.0 0.255.255.255 log
access-list 20 deny   220.0.0.0 3.255.255.255 log
access-list 20 deny   200.200.200.0 0.0.0.128 log
```

© SANS Institute 2000 - 2005, Author retains full rights.

```
access-list 20 permit any
access-list 50 permit 200.200.200.0 0.0.0.128
access-list 50 deny   any log
banner motd ^C
WARNING: Authorized access for GIAC system administrators only!
^C
!
line con 0
line aux 0
line vty 0 4
 password $1$Ohe0$lx2f2EX1Ts7T4l5ivuUE0.
 access-class 1 in
 login
!
end
```

Black Hole List:

Network/ Subnet	Owner/ Previous Owner	Date Assigned/ Revoked
000/8	IANA - Reserved	Sep 81
001/8	IANA - Reserved	Sep 81
002/8	IANA - Reserved	Sep 81
005/8	IANA - Reserved	Jul 95
007/8	IANA - Reserved	Apr 95
010/8	IANA - Private Use	Jun 95
014/8	IANA - Public Data Network	Jun 91
023/8	IANA - Reserved	Jul 95
027/8	IANA - Reserved	Apr 95
031/8	IANA - Reserved	Apr 99
036/8	IANA - Reserved	Jul 00
037/8	IANA - Reserved	Apr 95
039/8	IANA - Reserved	Apr 95
041/8	IANA - Reserved	May 95
042/8	IANA - Reserved	Jul 95
049/8	Joint Technical Command	May 94
	Returned to IANA	Mar 98
050/8	Joint Technical Command	May 94
	Returned to IANA	Mar 98
058/8	IANA - Reserved	Sep 81
059/8	IANA - Reserved	Sep 81
060/8	IANA - Reserved	Sep 81
069-079/8	IANA - Reserved	Sep 81
082-095/8	IANA - Reserved	Sep 81
096-126/8	IANA - Reserved	Sep 81
127/8	IANA - Reserved	Sep 81
197/8	IANA - Reserved	May 93
221-223/8	IANA - Reserved	Sep 81
224-239/8	IANA - Multicast	Sep 81
240-255/8	IANA - Reserved	Sep 81

© SANS Institute 2000 - 2005, Author retains full rights.

Firewall Config: PIX 525 Firewall running v6.1

```
: Saved
:
PIX Version 6.1(6)
nameif ethernet0 outside security0
nameif ethernet1 inside security100
nameif ethernet2 metaframe security70
nameif ethernet3 service security60
enable password QBL7X.xV9u1D5R9k encrypted
passwd QBL7X.xV9u1D5R9k encrypted
hostname giacpix
fixup protocol ftp 21
fixup protocol http 80
fixup protocol h323 1720
fixup protocol rsh 514
fixup protocol rtsp 554
fixup protocol smtp 25
fixup protocol sqlnet 1521
fixup protocol sip 5060
names
access-list acl_out permit tcp any host 200.200.200.10 eq 1494
access-list acl_out permit udp any host 200.200.200.10 eq 1494
access-list acl_out permit udp any host 200.200.200.10 eq 1605
access-list acl_out permit tcp any host 200.200.200.10 eq 81
access-list acl_out permit tcp any host 200.200.200.13 eq smtp
access-list acl_out permit tcp host 199.199.199.12 host 200.200.200.12 eq domain
access-list acl_out permit udp any host 200.200.200.12 eq domain
access-list acl_out permit udp host 128.118.25.3 any eq 123
access-list acl_out permit tcp any host 200.200.200.20 eq www
access-list acl_out permit tcp any host 200.200.200.17 eq www
access-list acl_out deny ip any any
access-list acl_service permit icmp any any
access-list acl_service permit udp host 10.202.0.2 any eq domain
access-list acl_service permit tcp host 10.202.0.3 any eq smtp
access-list acl_service permit tcp host 10.202.0.10 any eq www
access-list acl_service permit tcp host 10.202.0.7 any eq www
access-list acl_service permit udp any host 128.118.25.3 eq 123
access-list acl_service permit tcp host 10.202.0.2 host 199.199.199.12 eq domain
access-list acl_service permit tcp host 10.202.0.5 10.202.0.31 255.255.255.0 eq 22
access-list acl_service permit udp host 10.202.0.5 10.202.0.31 255.255.255.0 eq 22
access-list acl_service permit tcp host 10.202.0.5 host 10.202.0.20 eq 5432
access-list acl_service permit udp host 10.202.0.5 host 10.202.0.20 eq 5432
access-list acl_service permit tcp host 10.202.0.10 host 10.202.0.31 eq 22
access-list acl_service permit udp host 10.202.0.10 host 10.202.0.31 eq 22
access-list acl_service permit tcp host 10.202.0.10 host 10.202.0.31 range 137 139
access-list acl_service permit udp host 10.202.0.10 host 10.202.0.31 range netbios-ns 139
access-list acl_service deny ip any any
access-list acl_metaframe permit icmp any any
access-list acl_metaframe permit tcp any any eq 1494
access-list acl_metaframe permit udp any any eq 1494
access-list acl_metaframe permit udp any any eq 1605
access-list acl_metaframe permit tcp any any eq 81
```

© SANS Institute 2000 - 2005, Author retains full rights.

```
access-list acl_metaframe permit tcp any host 10.201.0.29 eq www
access-list acl_metaframe permit tcp any host 10.201.0.30 eq 524
access-list acl_metaframe permit udp any host 10.201.0.30 eq 524
access-list acl_metaframe permit tcp any host 10.201.0.22 range 137 139
access-list acl_metaframe permit tcp any host 10.201.0.23 range 137 139
access-list acl_metaframe permit udp any host 10.201.0.22 range netbios-ns 139
access-list acl_metaframe permit udp any host 10.201.0.23 range netbios-ns 139
access-list acl_metaframe permit tcp any host 10.201.0.20 eq 22
access-list acl_metaframe permit udp any host 10.201.0.20 eq 22
access-list acl_metaframe permit ip any host 10.201.0.1
access-list acl_metaframe deny ip any 10.201.0.0 255.255.255.0
access-list acl_in permit udp host 10.200.20.7 any eq 53
access-list acl_in permit ip 10.200.2.0 255.255.255.0 any
access-list acl_in permit tcp 10.200.2.0 255.255.255.0 host 10.200.0.1 eq 23
access-list acl_in deny tcp 10.200.0.0 255.255.0.0 host 10.202.0.10 eq 80
access-list acl_in permit ip host 10.200.20.5 any
access-list acl_in permit tcp host 10.200.3.1 host 10.202.0.5 eq 22
access-list acl_in permit tcp host 10.200.3.1 host 10.202.0.10 eq 22
access-list acl_in permit udp host 10.200.3.1 host 10.202.0.5 eq 22
access-list acl_in permit udp host 10.200.3.1 host 10.202.0.10 eq 22
access-list acl_in permit tcp host 10.200.3.1 host 10.202.0.10 range 137 139
access-list acl_in permit udp host 10.200.3.1 host 10.202.0.10 range 137 139
access-list acl_in permit tcp host 10.200.20.9 host 10.202.0.5 eq 5432
access-list acl_in permit udp host 10.200.20.9 host 10.202.0.5 eq 5432
access-list acl_in permit tcp host 10.200.20.10 host 10.201.0.2 eq 524
access-list acl_in permit udp host 10.200.20.10 host 10.201.0.2 eq 524
access-list acl_in permit tcp host 10.200.20.10 host 10.202.0.3 eq 25
access-list acl_in permit udp host 10.200.20.10 host 128.118.25.3 eq 123
access-list acl_in permit tcp host 10.200.20.2 host 10.201.0.2 range 137 139
access-list acl_in permit tcp host 10.200.20.3 host 10.201.0.2 range 137 139
access-list acl_in permit udp host 10.200.20.2 host 10.201.0.2 range 137 139
access-list acl_in permit udp host 10.200.20.3 host 10.201.0.2 range 137 139
access-list acl_in deny ip any any
pager lines 24
logging on
logging timestamp
no logging standby
no logging console
no logging monitor
logging buffered errors
no logging trap
no logging history
logging facility 3
logging queue 512
logging host service 10.202.0.8
logging host inside 10.200.20.8
interface ethernet0 auto shutdown
interface ethernet1 auto shutdown
interface ethernet2 auto shutdown
interface ethernet3 auto shutdown
mtu outside 1500
mtu inside 1500
ip address outside 200.200.200.2 255.255.0.0
ip address inside 10.200.0.1 255.255.255.128
```

© SANS Institute 2000 - 2005 Author retains full rights.

```
ip address metaframe 10.201.0.1 255.255.255.128
ip address service 10.202.0.1 255.255.255.0
ip audit info action alarm
ip audit attack action alarm
arp timeout 14400
global (outside) 1 200.200.200.3 netmask 255.255.255.255
global (service) 1 10.202.0.50 netmask 255.255.255.255
global (metaframe) 1 10.201.0.50 netmask 255.255.255.255
nat (inside) 1 0.0.0.0 0.0.0.0 0 0
static (inside,metaframe) 10.201.0.30 10.200.20.10
static (inside,metaframe) 10.201.0.22 10.200.20.2
static (inside,metaframe) 10.201.0.23 10.200.20.3
static (inside,metaframe) 10.201.0.20 10.200.20.20
static (inside,metaframe) 10.201.0.29 10.200.20.9
static (metaframe,outside) 200.200.200.10 10.201.0.2
static (service,outside) 200.200.200.12 10.202.0.2
static (service,outside) 200.200.200.13 10.202.0.3
static (service,outside) 200.200.200.17 10.202.0.7
static (service,outside) 200.200.200.20 10.202.0.10
static (inside,service) 10.202.0.20 10.202.0.9
static (inside,service) 10.202.0.31 10.200.3.1
access-group acl_out in interface outside
access-group acl_in in interface inside
access-group acl_metaframe in interface metaframe
access-group acl_service in interface service
route outside 0.0.0.0 0.0.0.0 200.200.200.1 1
timeout xlate 3:00:00
timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00 rpc 0:10:00 h323 0:05:00 si
p 0:30:00 sip_media 0:02:00
timeout uauth 0:05:00 absolute
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
no snmp-server location
no snmp-server contact
snmp-server community caig-etavirp
no snmp-server enable traps
floodguard enable
no sysopt route dnat
isakmp identity hostname
telnet 10.200.2.0 255.255.255.0 inside
telnet timeout 15
ssh timeout 5
terminal width 80
Cryptochecksum:9a99baeeecdc9d4c9d9fe3921ea676ff
```

Suggested Re-Ordering of PIX ACL's:

```
access-list acl_out permit tcp any host 200.200.200.20 eq www
access-list acl_out permit udp any host 200.200.200.12 eq domain
access-list acl_out permit udp host 128.118.25.3 any eq 123
access-list acl_out permit tcp any host 200.200.200.13 eq smtp
access-list acl_out permit tcp any host 200.200.200.17 eq www
access-list acl_out permit udp any host 200.200.200.10 eq 1494
```

© SANS Institute 2000 - 2005, Author retains full rights.

```
access-list acl_out permit tcp any host 200.200.200.10 eq 1494
access-list acl_out permit tcp host 199.199.199.12 host 200.200.200.12 eq domain
access-list acl_out permit udp any host 200.200.200.10 eq 1605
access-list acl_out permit tcp any host 200.200.200.10 eq 81
access-list acl_out deny ip any any
access-list acl_service permit tcp host 10.202.0.10 any eq www
access-list acl_service permit udp host 10.202.0.2 any eq domain
access-list acl_service permit udp any host 128.118.25.3 eq 123
access-list acl_service permit tcp host 10.202.0.3 any eq smtp
access-list acl_service permit tcp host 10.202.0.7 any eq www
access-list acl_service permit tcp host 10.202.0.5 10.202.0.31 255.255.255.0 eq 22
access-list acl_service permit udp host 10.202.0.5 10.202.0.31 255.255.255.0 eq 22
access-list acl_service permit tcp host 10.202.0.5 host 10.202.0.20 eq 5432
access-list acl_service permit udp host 10.202.0.5 host 10.202.0.20 eq 5432
access-list acl_service permit tcp host 10.202.0.10 host 10.202.0.31 eq 22
access-list acl_service permit udp host 10.202.0.10 host 10.202.0.31 eq 22
access-list acl_service permit tcp host 10.202.0.10 host 10.202.0.31 range 137 139
access-list acl_service permit udp host 10.202.0.10 host 10.202.0.31 range netbios-ns 139
access-list acl_service permit tcp host 10.202.0.2 host 199.199.199.12 eq domain
access-list acl_service permit icmp any any
access-list acl_service deny ip any any
access-list acl_metaframe permit icmp any any
access-list acl_metaframe permit tcp any any eq 1494
access-list acl_metaframe permit udp any any eq 1494
access-list acl_metaframe permit tcp any host 10.201.0.30 eq 524
access-list acl_metaframe permit udp any host 10.201.0.30 eq 524
access-list acl_metaframe permit tcp any host 10.201.0.22 range 137 139
access-list acl_metaframe permit tcp any host 10.201.0.23 range 137 139
access-list acl_metaframe permit udp any host 10.201.0.22 range netbios-ns 139
access-list acl_metaframe permit udp any host 10.201.0.23 range netbios-ns 139
access-list acl_metaframe permit tcp any host 10.201.0.20 eq 22
access-list acl_metaframe permit udp any host 10.201.0.20 eq 22
access-list acl_metaframe permit tcp any host 10.201.0.29 eq www
access-list acl_metaframe permit udp any any eq 1605
access-list acl_metaframe permit tcp any any eq 81
access-list acl_metaframe permit ip any host 10.201.0.1
access-list acl_metaframe permit icmp any any
access-list acl_metaframe deny ip any 10.201.0.0 255.255.255.0
access-list acl_in permit ip 10.200.2.0 255.255.255.0 any
access-list acl_in deny tcp 10.200.0.0 255.255.0.0 host 10.202.0.10 eq 80
access-list acl_in permit ip host 10.200.20.5 any
access-list acl_in permit udp host 10.200.20.7 any eq 53
access-list acl_in permit udp host 10.200.20.10 host 128.118.25.3 eq 123
access-list acl_in permit tcp host 10.200.20.10 host 10.202.0.3 eq 25
access-list acl_in permit tcp host 10.200.3.1 host 10.202.0.5 eq 22
access-list acl_in permit tcp host 10.200.3.1 host 10.202.0.10 eq 22
access-list acl_in permit udp host 10.200.3.1 host 10.202.0.5 eq 22
access-list acl_in permit udp host 10.200.3.1 host 10.202.0.10 eq 22
access-list acl_in permit tcp host 10.200.3.1 host 10.202.0.10 range 137 139
access-list acl_in permit udp host 10.200.3.1 host 10.202.0.10 range 137 139
access-list acl_in permit tcp host 10.200.20.10 host 10.201.0.2 eq 524
access-list acl_in permit udp host 10.200.20.10 host 10.201.0.2 eq 524
access-list acl_in permit tcp host 10.200.20.9 host 10.202.0.5 eq 5432
access-list acl_in permit udp host 10.200.20.9 host 10.202.0.5 eq 5432
```

© SANS Institute 2000 - 2005, Author retains full rights.

```
access-list acl_in permit tcp host 10.200.20.2 host 10.201.0.2 range 137 139
access-list acl_in permit tcp host 10.200.20.3 host 10.201.0.2 range 137 139
access-list acl_in permit udp host 10.200.20.2 host 10.201.0.2 range 137 139
access-list acl_in permit udp host 10.200.20.3 host 10.201.0.2 range 137 139
access-list acl_in permit tcp 10.200.2.0 255.255.255.0 host 10.200.0.1 eq 23
access-list acl_in deny ip any any
```

© SANS Institute 2000 - 2005, Author retains full rights.

References (in order of use):

Microsoft's DNS Problems Press Release, 2001: http://www.microsoft.com/presspass/press/2001/Jan01/01-24DNSpr.asp

WebSense Home Page: http://www.websense.com/

Cisco PIX Throughput Information: http://www.cisco.com/warp/public/cc/pd/fw/sqfw500/

RFC 1918, Address Allocation for Private Internets, February 1996: http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc1918.html

Craig A. Huegen, The Latest in Denial of Service Attacks: "SMURFING" Description and Information To Minimize Effects, February 8, 2000: http://www.pentics.net/denial-of-service/white-papers/smurf.cgi

Configuration Guide for the Cisco Secure PIX Firewall Version 6.0: http://www.cisco.com/univercd/cc/td/doc/product/iaabu/pix/pix_60/config/config.htm

CERT Advisory CA-2002-03 Multiple Vulnerabilities in Many Implementations http://www.securityfocus.com/archive/1/255807

John the Ripper password cracking utility: http://www.openwall.com/john/

Internet Assigned Numbers Authority, Port Numbers, updated January 25, 2002: http://www.iana.org/assignments/port-numbers

NMAP Stealth Port Scanner Home Page: http://www.insecure.org/nmap

Tcpdump Home Page: www.tcpdump.org

Windump Home Page: www.windump.org

PIX **alias** command: http://www.cisco.com/univercd/cc/td/doc/product/iaabu/pix/pix_61/cmd_ref/a.htm#65183

Daniel S. Martin, GIAC Certified Firewall Analyst Practical, March 28, 2001: http://www.giac.org/practical/Daniel_Martin_GCFW.doc

F-Secure Subseven Information: http://www.europe.f-secure.com/v-descs/subseven.shtml

Symantec Security Response: http://securityresponse.symantec.com

W97M.Pacol.A Details: http://securityresponse.symantec.com/avcenter/venc/data/w97m.pacol.a.html

W32.Klez.E@mm Details: http://securityresponse.symantec.com/avcenter/venc/data/w32.klez.e@mm.html

Microsoft IIS 5.0 .printer ISAPI Extension Buffer Overflow: http://www.securityfocus.com/bid/2674

Most of the information presented in this paper without a reference, such as default policies, common configurations, and not-so-common sense, was learned from colleagues at work or in the field. In no particular order, special thanks to: Chris Benton, Scott Bookmiller, Bill Browder, Johan Greefkes, Bob Hirsch, Jason Killam and Bill Smith. Many thanks to those not mentioned whose contributions are not forgotten.

© SANS Institute 2000 - 2005, Author retains full rights.