# Global Information Assurance Certification Paper

# GCFW Practical

Version 4.1



Mike Jensen

Submitted: 12/19/2004

## Abstract

GIAC Enterprises is a small business which markets fortune cookie sayings to customers worldwide.[1] (giac.org, p. 2) This work describes the security architecture used by GIAC. Also presented is an essay regarding Host Intrusion Prevention Systems, including some drawbacks and benefits to such a system and the potential impact such a system could have on the GIAC network. Although fortune cookies are ubiquitous and freely distributed at nearly every Chinese restaurant, GIAC Enterprises does not have limitless cash reserves. However, security is a major priority and has been emphasized. The network design reflects fiscal responsibility with an eye towards future expansion and maximum security for the price.

## Assignment 1

To understand Host Intrusion Prevention Systems (HIPS), it is useful to first understand Host Intrusion Detection Systems (HIDS). HIDS can be as simple as a custom perl script that tails a log file, watching for specific events. HIDS can be as complex as a commercial product with a management framework, event correlation engine and sophisticated alerting system. All of these share a common purpose; to be aware of security events taking place on a specific system. As the name implies HIDS is the monitoring of a system for signs of intrusion, or unauthorized access or use. Some products accomplish this by watching the system and log files that a host generates for suspicious events as defined by signatures.[2] (SANS p. 1-29) Others monitor binary file sizes and process memory sizes for changes that could indicate foul play.[3] (SANS p. 1-28) Some products combine these. But at the end of the day these devices are passive. They serve to detect intrusions only and they don't even do it in real-time. An event must occur before the HIDS can alert on it. It would be similar to a security guard at a jewelry store watching a robbery and giving a very detailed description of the perpetrators to the police. It's nice to know exactly what happened and who they were, but it would have been even nicer to prevent the robbery in the first place. This same desire gave rise to the creation of HIPS.

There are similar technologies on the network-based side of the house. Network Intrusion Detection Systems (NIDS) and Prevention Systems (NIPS) evolved in a similar manner. However, it was a conceptually simple matter to convert from passively monitoring a network connection to actively denying a network connection. One answer was to give the NIDS system the ability to generate packets that could pretend to be from the systems involved in the network transaction and reset that connection. This worked pretty well but tended to miss events in a high traffic situation. Another solution was to place the NIPS "inline." This meant that a network bottleneck was either chosen or created, and a NIPS device was placed in the middle. For traffic to get to one side or the other it had to traverse the NIPS successfully. This prevented the NIPS from missing anything but also provided a potential point for network slowness. Because traffic was forced to flow through this point the network

could only flow as quickly as the slowest NIPS.

These concepts were a natural extension of what NIDS were already doing, but how could these same concepts be incorporated into HIPS? HIDS were in the business of log files. Simply preventing a log entry from being written would not prevent the event. One method of event prevention was giving the HIDS control over system calls.[4] (SANS p. 1-30) Whenever an application requires system resources, it makes a system call for them. This is a nearly analogous concept to NIPS, where requests for network resources consist of network calls for those resources. Giving HIPS control of the system calls on a machine provides that bottleneck that an inline NIPS system needs to prevent events slipping by.

This method works by defining signatures for the expected behavior of an application. If a system call is made by an application that falls outside of the expected behavior, the HIPS denies it.[5] (SANS p. 1-30) This means that any spurious program or behavior change to an approved program cannot function on a HIPS protected system. It also means that the HIPS has to be very intelligent about the application you want to run. Consider the example of a server process that is vulnerable to a buffer overflow. A malicious attacker finds this server process and launches the exploit recently downloaded from the Internet against it. On an unprotected server the attacker now has control of a running server process and can execute code in the context of that process. On a HIPS-protected server the attacker has control of that same process, but any actions that the attacker attempts that are outside of the normal, expected behavior simply fail. By the same token consider a well-intentioned server administrator applying the latest update to a HIPS-protected server under his care. The upgrades are applied but the server admin forgets to also update the signatures on the HIPS to accompany the new software version. The freshly upgraded software keeps crashing, and until the server admin remembers that this system has HIPS installed it will probably be very difficult to troubleshoot.

Some benefits of HIPS are the control that this gives over a system. Assuming that the HIPS is functioning and configured correctly, nothing can take place on a system that is not sanctioned by the HIPS. This goes to the root of many security problems. The number of bank robbers cruising around a city is of no interest if every bank in town is impenetrable. Also, in these times of encrypted traffic streams, Gramm-Leach-Bliley and VPNs, sometimes the only place to get a good look at what is happening in these network transactions is on one of the endpoints, after that stream is decrypted. A network-based intrusion detection or prevention device can see that an SSL-encrypted HTTP stream is flowing by, but can't tell what is inside that stream.[6] (SANS p. 1-32) However, HIPS running on the web server at the end of the stream can see what is going on.

Drawbacks to this type of application are that many times the assumption that the HIPS is functioning and configured correctly will not be a good one. Revisiting the example above, the chances of every bank in town being impenetrable are extremely low. HIPS protection may mean that the choice in what server software can be deployed to a system is limited to those packages

that are supported by the particular HIPS product. This can have serious consequences on the future of a system. If the usage requirements for a server move in a direction that the HIPS provider has not yet gone or has no intention of going, a decision must be made to either abandon the HIPS or the usage direction. Also, vulnerabilities in a HIPS product could have some extremely serious consequences. Because HIPS controls system calls, an attack that disables HIPS could completely disable the system. If control of the HIPS is gained, the attacker has gained the "keys to the kingdom," capable of anything on the system. At the very least disabling HIPS blinds a system to potential second-stage attacks, perhaps allowing these to continue. A HIPS system could give a server administrator a lax attitude in applying system patches, instead relying on the protection afforded by the HIPS.

Applying the concepts of HIPS to the GIAC Enterprises security architecture could make a lot of sense. A large portion of GIAC's business is conducted through their web site, utilizing a lot of SSL encryption. The network-based IDS sensors would not be able to see inside these SSL connections. HIPS running on the web server would be able to examine these connections, however. The placement of the web server in the architecture would also reduce the exposure of the HIPS to attack. The only access to the web server from anyone other than a web administrator is through at least one proxy firewall and a reverse web proxy on the DMZ subnet.

In terms of Defense-in-Depth, HIPS would provide one more safety net. In the event that an attack makes it through the filtering router, past the primary firewall, through the reverse proxy's traffic validation, past the secondary firewall and to the web server, if that attack doesn't adhere to strict guidelines defined for the server processes on that system, it will fail. Or, if HIPS were installed on the SSH server located in the DMZ. An attack that originates from one of the predefined sources that overcomes the requirement for public-private key pairs and successfully exploits an unpatched vulnerability in the SSH server software would be limited only the actions that are predefined for OpenSSH.

There would definitely be value in adding HIPS to the GIAC network. But the decision would be made on cost versus benefit. If the HIPS system were inexpensive, perhaps freely downloadable open-source and didn't require a dedicated employee to administer or a skill set that an existing employee lacks, it would make a lot of sense. It would also have to support the platforms and applications that are already deployed in the architecture. However, if these requirements weren't met, the added layers of security would probably not justify the added cost. There are many layers of security already in place. More security is always a good thing, but the risk must be weighed against the reward.

## Assignment 2

In general, the GIAC Enterprises (GIAC) network will be divided into four distinct sections - a service network (or DMZ), a server subnet, a user subnet and an Intrusion Detection System (IDS) monitoring subnet. These sections will be separated from each other and the Internet by firewalls.

As each group has different requirements of their interactions with GIAC Enterprises, there isn't a single access method that will work for each group. Below those interactions are briefly described for each group.

**Customers**

GIAC customers require access to an online catalog, an interface to place orders, a secure method to pay or arrange payment for ordered product, and a method to obtain or download the purchased product. These requirements lend themselves well to a web-based application. This application would require a combination of HTTP and HTTPS traffic from the customer to the entry point to GIAC's network. One way to implement this would be to place a web server in the DMZ and allow customer connections in through the firewall to this server. However, this would leave the web server open to the Internet on HTTP/HTTPS ports (typically 80/tcp and 443/tcp), which is generally a risky proposition. To mitigate this risk, our DMZ will host a reverse proxy. This device doesn't really serve its own content, but validates the request for content and then passes that request on to the web server located in the server subnet. This will help to insulate the web server from attacks that are RFC compliant and appear to be "normal" web traffic by putting an HTTP-aware device in the path for validation.

The online catalog functions, order interface, purchasing functions and download functions will all be incorporated into the back-end application that will make calls to the database server as needed.

**Suppliers**

GIAC suppliers are delivering fortune cookie sayings, which are primarily text. Working from the premise that time is money, and time spent uploading a text file is just as valuable, the assumption has been made that the suppliers are delivering as small a file as possible to reduce transfer times and storage requirements. Practically this takes the form of a compressed text file, comma delimited for ease of adaptation to a variety of formats. Requirements for delivering this file are fairly simple, consisting of a server to put it on and a secure way to put it there. For this GIAC is using an SSH server located in the DMZ. GIAC have provided the suppliers with an account on this server and the suppliers have provided a public key to help guarantee the connection source. The suppliers will put the file (using secure copy, or scp) in a predetermined

directory on the SSH server. GIAC can now retrieve the file from this location. The firewall will only allow SSH (22/tcp) connections to this server.

**Partners**

Partners share many of the same requirements and considerations as customers. The difference lies in what the partners do with the fortunes once obtained. Due to this, the partner interface will be very similar to the customer's. Partners will access the functions of the online fortune catalog via the reverse proxy with the same security considerations as customers.

**Internal employees**

Currently internal employees represent a greater threat to information security than external attackers. This effect is likely less pronounced in a small company like GIAC, but still present. Therefore the user subnet is segregated from the rest of the network by a firewall. This helps to ensure that a web server administrator can access the servers she is responsible for, but has no access to the payroll servers. Likewise the payroll personnel cannot access the web server inappropriately. Generally users with a business need for specific access to a system in the DMZ, server subnet or IDS monitoring subnet will be granted exactly what is needed, no more or less.

**Remote users**

GIAC remote users are basically internal employees who are external. They need the same sort of access and threat mitigation that internal employees need, but this must be applied remotely. To accomplish this GIAC will deploy an SSL VPN device to the DMZ. The SSL VPN device has the benefit of needing little more than a reasonably current web browser and a connection to the Internet. This means that a remote user can gain access from a broad variety of sources in a secure, encrypted manner. Each remote user is given a Windows XP workstation on the user subnet. The primary firewall allows HTTP/HTTPS traffic to the SSL VPN. The VPN uses RADIUS to authenticate the user. Once authenticated the remote user has access to connect via Remote Desktop (3389/tcp) to his workstation on the user subnet. From this point required access to servers and systems is provisioned as if the user were on the internal network.

For remote offices (regional offices dispersed globally) persistent VPN connections will be used. Each regional office will have an inexpensive high-speed Internet connection like DSL or cable modem. A Raptor VPN device will connect to this and establish a VPN tunnel with the primary firewall at GIAC's home office. Again each remote user will be given a Windows XP workstation on the user subnet. Users in the regional office will use Remote Desktop to connect to their workstations, and access permissions will be granted accordingly.

**General public**

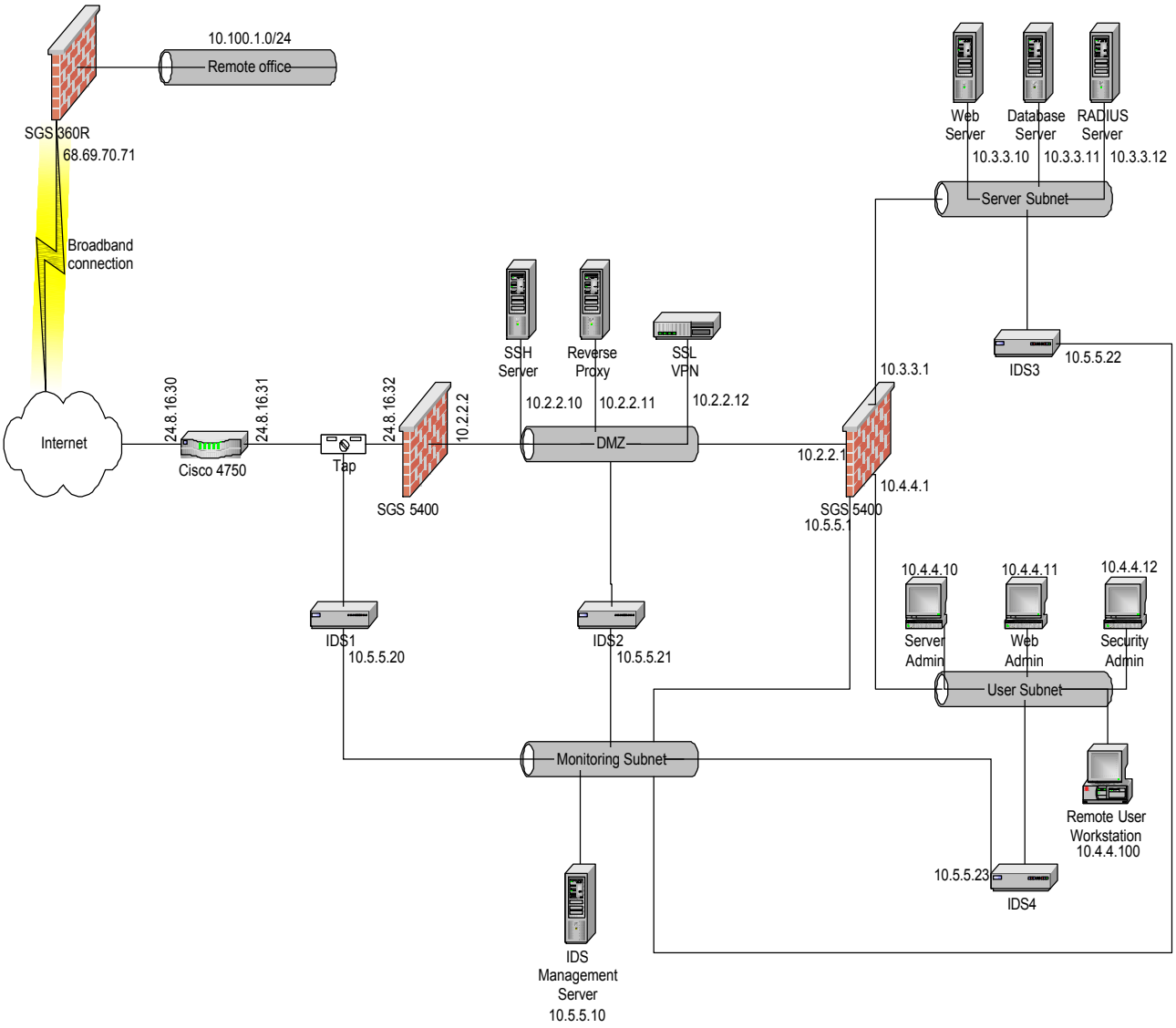
The general public will mostly require a simple website to find information

about GIAC, employment opportunities, etc. This will require similar access as customers and partners from a security standpoint, but without much of the back-end database requirements. These requests will first go through the reverse proxy for validation before being passed to the web server.

| User group | Source | Destination | Port (Protocol) | Description |
|---|---|---|---|---|
| Customers | Customer | Reverse proxy | 80/tcp (HTTP)<br>443/tcp (HTTPS) | Customer access to online catalog |
| | Reverse proxy | Web server | 80/tcp (HTTP)<br>443/tcp (HTTPS) | Second part of customer request |
| | | | | |
| Suppliers | Supplier server | SSH server | 22/tcp (SSH) | Supplier delivery of sayings |
| | | | | |
| Partners | Partner | Reverse proxy | 80/tcp (HTTP)<br>443/tcp (HTTPS) | Partner access to sayings |
| | Reverse proxy | Web server | 80/tcp (HTTP)<br>443/tcp (HTTPS) | Second part of partner access |
| | | | | |
| Employees (internal) | Web server administrator | Web server | 22/tcp (SSH)<br>80/tcp (HTTP)<br>443/tcp (HTTPS) | Admin access to web server |
| | Server administrator | SSH server | 22/tcp (SSH) | Admin access to SSH server |
| | Server administrator | Reverse proxy | 22/tcp (SSH) | Admin access to reverse proxy |
| | Database administrator | Database server | 22/tcp (SSH)<br>3306/tcp (MySQL) | Admin access to database server |
| | Security administrator | Firewalls | 423/tcp (command line interface)<br>2456/tcp (graphical firewall management tool) | Symantec management utilities, command line and graphical |
| | Security administrator | Monitoring subnet | 22/tcp (SSH) | Access to IDS subnet |
| | Security administrator | IDS Management server | 80/tcp (HTTP)<br>443/tcp (HTTPS) | Access to view and manage IDS alerts |
| | Security administrator | Router | 23/tcp (telnet) | Router management |
| | General employee | Reverse proxy | 80/tcp (HTTP)<br>443/tcp (HTTPS) | Employee access to online catalog |

| | | | | |
|---|---|---|---|---|
| | General employee | Secondary firewall | 8080/tcp (HTTP proxy) | Employee Internet access |
| | Secondary firewall | Internet | 80/tcp (HTTP)<br>443/tcp (HTTPS) | Access through primary firewall for employee web access |
| | Server administrator | SSH Server | 22/tcp (SSH) | Access to move the sayings delivered by suppliers off of the SSH server |
| | Server administrator | Database server | 22/tcp (SSH) | Access to move the sayings from suppliers on to the database server for importation |
| | | | | |
| Employees (remote) | Remote user | SSL VPN | 80/tcp (HTTP)<br>443/tcp (HTTPS) | Remote access to SSL VPN |
| | SSL VPN | RADIUS server | 1645/tcp (RADIUS) | Authentication requests for SSL VPN |
| | SSL VPN | User subnet | 3389/tcp (Remote Desktop) | Authenticated user access to XP workstations |
| | Remote SGS 360R | Primary firewall | 500/udp (ISAKMP) | ISAKMP key negotiation to establish tunnel |
| | Remote SGS 360R | Primary firewall | 50/ip (IPSEC encapsulation) | Encapsulation of encrypted tunnel |
| | Remote VPN tunnel | User subnet | 3389/tcp (Remote desktop) | Remote office access to XP workstations |
| | | | | |
| General public | Internet | Reverse proxy | 80/tcp (HTTP)<br>443/tcp (HTTPS) | General access to website |
| | Reverse proxy | Web server | 80/tcp (HTTP)<br>443/tcp (HTTPS) | Second half of website access |

**Data flow table 1**




**Security Architecture Diagram 1**

Taken separately every security component has weaknesses that could potentially be exploited to circumvent that component. The concept of Defense-in-Depth seeks to mitigate these individual threats by layering security components such that the strengths of one component compensate for the weaknesses of another. This section discusses the individual components of GIAC's design and how they complement each other to provide a secure environment.

The router is the only security component that physically connects to the Internet. Its main purpose is to route traffic between the Internet and the GIAC network, and filter that traffic using ingress and egress access control lists (ACLs). These ACLs will prevent our internal traffic from traversing the Internet inappropriately in the event of a misconfiguration, shield our firewall from unnecessary threats, and provide a good place to deal with certain types of attacks like denial-of-service (DoS) attacks from specific sources. Because it is the first component of the network that Internet-sourced traffic encounters, it is an ideal place to drop traffic that we have no use for. For example, GIAC is not offering the ability to telnet to any of the servers on the DMZ subnet. Therefore, there is no need to allow port 23/tcp into the DMZ from the Internet. This traffic can be dropped at the router.

The router's position as final component before internal traffic reaches the Internet makes it an ideal location for "last resort" filtering, such as preventing "private" IP addresses from reaching the Internet. An example of this could be a filter restricting ports 135/tcp – 139/tcp outbound. In the event of a file-share virus outbreak on the GIAC network, this filter could prevent the virus from spreading to other networks across the Internet.

A Cisco 3745 was chosen for this component. It is a fairly popular mid-range router with all of the options required, as well as room for expansion in the future.

Routers are not so much security devices as they are network devices with security functions. They have been designed to pass traffic first, and then worry about security. In the event of problems with these routers the can fail "open," so that as little as possible will interfere with their ability to pass traffic. In the past vulnerabilities in Cisco IOS have caused the router to fail, potentially leading to such a situation.[7] (Security Focus BID 4132) Other issues have allowed unauthorized users inappropriate levels of access to the router.[8] (Security Focus BID 2936) In the GIAC design these potential threats are mitigated by the presence of the firewall directly behind the router and a network-based Intrusion Detection System (IDS). The firewall denies most traffic, allowing only what its policy defines as necessary. Any traffic that the router should block should also be blocked by the firewall. The IDS inspects the traffic to ensure that the router is behaving appropriately, sending an alert if this is not the case.

The primary firewall's purpose is to protect the network behind it from potentially dangerous traffic, as well as pass required traffic in a safe way. The firewall chosen is the Symantec Gateway Security 5400 firewall appliance. It is a true proxy firewall providing a high degree of application awareness and verification. It also provides the capability to initiate and terminate IPSEC VPN tunnels, which will be required for connectivity to the GIAC regional offices. These appliances can be incorporated into a cluster, providing room for expansion when necessary.

One weakness with a proxy firewall is that in order for it to proxy traffic, it must bind a daemon to whatever ports it intends to pass traffic on. For example,

if the firewall policy allows HTTP traffic to pass in to the DMZ, the firewall must have a daemon bound to port 80/tcp. Vulnerabilities in these listening daemons could result in security breaches of the firewall. Considering that firewall processes typically run in the context of "super-user," those breaches could be very serious.

The router described above helps mitigate this weakness. The router will shield the firewall from any ports that are not strictly required. If a source address is known to be problematic, the router can deny traffic from this address. An IDS system is deployed to monitor traffic bound for this firewall. If this system sees traffic that is potentially hazardous to the firewall it will alert. But the most important mitigating factor is a security/firewall administrator who makes sure the firewall is at current patch levels. A properly patched and maintained proxy firewall reduces the danger of unpatched servers on the internal network behind it.

The secondary firewall is also a Symantec SGS 5400 appliance. In addition to providing firewall features, this device is also capable of serving as a web proxy with HTTP anti-virus capabilities. This protects workstations using it from downloading viruses hidden on infected web servers, like the Nimda virus.[9] (McAfee) Using the same firewall here as the primary firewall also reduces the management burden by standardizing on a single firewall platform.

A Symantec SGS 360R Firewall/VPN appliance will protect each remote office. This device is relatively inexpensive ($850 USD)[10] (Symantec Store) but includes many nice features like integrated intrusion detection/prevention, content filtering of web traffic, and remote management. In addition, this device will create a gateway-to-gateway IPSEC VPN tunnel with the primary firewall at the main office.

Weaknesses with this device lie in the fact that it is a stateful packet inspection firewall. Packets meeting the requirements of the firewall policy will be routed through to the network behind. To mitigate this the only inbound access allowed will be from the external address of the main office (24.8.16.32) to the device itself on ports 80/tcp and 443/tcp, for remote management. The only traffic allowed to traverse the VPN tunnel will be 3389/tcp (Remote Desktop) to the user subnet, where each remote office user will have a Windows XP workstation for their use. This will help limit what a system in the remote office that has been infected with a virus or compromised by an attacker can do across the encrypted tunnel. Should something like compromised hosts occur, password strength and security will be a large factor in limiting the damage done. For this reason password policies will include minimum lengths of 8 characters, a mixture of alpha, numeric and special characters, and a 90-day expiration.

The SSH server is OpenSSH 3.9p1 running on FreeBSD 5.3. Both the operating system and SSH server are open-source, free software, reducing the cost of the server. FreeBSD was chosen as the standard operating system because of its network performance and its capability of performing all the

required server responsibilities, and its cost. OpenSSH was chosen because of its cost and its long history of security and rapid availability of patches in the event of a vulnerability.

This component's purpose is to provide a secure environment for interaction between GIAC's suppliers and GIAC. The primary potential weakness with this piece is that it is a listening server with some interaction with the Internet. To mitigate this threat public-private key pairs will be utilized to verify that incoming connections have the right to attempt authentication. The firewall policy will only allow port 22/tcp connections to the server from the Internet. The server administrator will maintain appropriate patch levels on the server. The server has been placed in the DMZ to further mitigate the possibility it could become compromised, perhaps by a new, unpublished exploit. The secondary firewall policy will deny connections from this server in to the internal network to limit the damage it could do were it compromised. An IDS sensor is monitoring the mirror port to alert the security administrators to "unexpected" traffic patterns that could indicate a security breach.

A large component of GIAC's business will be conducted through its web server. Historically web servers can be a risky component of a network. Because of these factors, GIAC will use a reverse proxy to provide a layer of "insulation" between the Internet and the web server. The reverse proxy accepts the initial connection and only forwards the request on if it is of an allowed type. For example, an attempt to execute a directory traversal will be specifically denied at the reverse proxy.

This reverse proxy consists of Apache 2.0.52 on FreeBSD 5.3. FreeBSD is our standard operating system. Apache was chosen because of its cost and its reputation as a secure web server. It is an open-source product and is actively maintained and constantly reviewed. This has produced a safe and stable product.

The primary weakness in this component is that it does accept connections from the Internet and therefore could become compromised should an attacker exploit a vulnerability in the web server. In answer to this, the primary firewall only allows connections to this server on ports 80/tcp and 443/tcp from the Internet. In addition the proxy firewall performs traffic validation to ensure that the connections bound for this server are RFC compliant. Unfortunately this doesn't eliminate the risk; it is possible that exploits could be contained in "normal" web traffic. This is part of the reason that a reverse proxy is being used. The IDS on the mirror port for this subnet will watch the traffic for exploit attempts that may be within RFC confines yet still present a risk to the server. The only connections allowed past the secondary firewall are to the web server on ports 80/tcp and 443/tcp so that, in the event that the reverse proxy is compromised it is limited in what it can do to the systems on the internal network.

One more threat vector lies in systems that it shares the DMZ with. The primary method of administration for this system will be via SSH from the user subnet. For this reason the reverse proxy will have OpenSSH 3.9p1 installed

and bound to port 22/tcp. Should the SSH server become compromised, it will be through OpenSSH because the firewall policy prevents connections to other services. Because both servers are running the same version of OpenSSH, if one is susceptible, they both are. An attacker that successfully gains entry to the SSH server could use it as an attack platform to compromise the reverse proxy. Breaching this box gains the attacker access to the web server on ports 80/tcp and 443/tcp. To mitigate this threat the IDS on the mirror port specifically monitors for SSH connections from the SSH server to the reverse proxy and alerts the security administrators should one occur. Under normal circumstances no one should attempt this type of connection.

A remote, traveling sales force represents an interesting challenge in providing VPN services. One day they may be connecting from a client's network, that evening from their hotel room, the next day a public wireless access point at the airport. VPN client software can meet this need, but only if the salesperson has access to their laptop at all times. By using an SSL VPN, a remote user can gain secure network access from any machine that has a 128-bit SSL-capable browser. If that machine also has the Microsoft Remote Desktop client, the remote user has the same abilities as if in the office. This technology is useful not only to the traveling sales force, but also to employees away from the office. The occasional emergency page while at a friend's Christmas party doesn't mean that a server administrator has to drive in to work, or even home where she has the VPN client installed. If the friend has an Internet connection and a browser many times crisis can be averted using an SSL VPN.

The product chosen for GIAC's network is the Juniper Networks Netscreen-SA 1000 Series appliance. This product supports more than enough users and has a high degree of configurability, including the ability to grant users access to different resources based on the URL requested. It supports RADIUS authentication, which is what GIAC will use. Its weaknesses are very similar to the reverse proxy or any other web server. Because it accepts connections from the Internet on port 80/tcp and 443/tcp, it is potentially vulnerable to malicious traffic on these ports. Mitigation is similar to the reverse proxy. The proxy firewall provides RFC and traffic checking prior to requests reaching the device. The server administrator will maintain appropriate patch levels. The secondary firewall policy only allows connections from the SSL VPN to the RADIUS server on the server subnet on port 1645/tcp, and to the user subnet on port 3389/tcp, which is required for Microsoft Remote desktop (RDP.)

This device has been placed in the DMZ because it does accept connections from the Internet. Should an unpublished or previously unknown exploit be successful against it, there are limited options that the attacker has open. Once control of the SSL VPN was gained more research would have to be done to try and find another exploit or hole to do any more damage. This research would likely be detected by IDS.

Each employee who will connect remotely will have a Windows XP workstation on the user subnet. Employees at the home office will use their

standard workstation. The remote sales force will have a specific workstation assigned to them for remote access. One reason this is done is to simplify the provisioning of access privileges. A web server administrator only needs access permissions from his workstation to the web server. He doesn't need additional permissions from the SSL VPN to the web server. It also reduces the access required through the secondary firewall from the SSL VPN.

Preventative security measures like firewalls are a requirement for a secure network, but they're not enough. Banks with a lot of currency on the premises use strong preventative physical security measures like vaults and alarms to protect that currency. But they also use security guards to protect against the unexpected weaknesses, the betrayal by "trusted" employees, or the exploitation of necessary access policies. A dynamic threat landscape requires dynamic threat response, and security measures require verification of success. An intrusion detection system is the security guard of a network. Where potentially risky access must be granted for proper function, as in the case of our reverse proxy above, the IDS watches to alert if someone tries something potentially harmful. Where an acceptable traffic pattern is known, the IDS watches for deviations from this known pattern. When a new exploit is released before the vendor of the affected system can provide a patch, the IDS can quickly be configured to alert us to someone trying this new exploit, provided enough information about the exploit is available. This can allow us to react appropriately, perhaps by blocking the attacker's IP address at our filtering router.

GIAC will use Snort 2.2.0 on FreeBSD 5.3. FreeBSD is our standard non-Windows operating system, but is especially suited to an IDS application. A network-based IDS sensor's main purpose in life is to look at as much traffic as possible, hopefully every packet that passes by. The network stack in FreeBSD is very robust and capable of handling high network loads. Snort is open-source IDS software developed by Martin Roesch. It is an extremely popular IDS software, it's free, and it is easy to create custom signatures. There is a very large community that uses Snort, providing plentiful support and rapid patches and new signatures. Since the creation of Sourcefire, a company started in part by Martin Roesch to make Snort sensors commercially available, the community support has gotten even better. Now some of the best Snort developers actually get paid for their work, allowing them to devote more time to developing. All of these factors combine to make Snort a suitable choice for GIAC's IDS.

Each sensor has two network connections, one on the monitoring subnet, and a "promiscuous" one in each monitoring location. This promiscuous connection does not have an IP address associated to it, and is essentially "invisible" to the network segment it is monitoring. This greatly reduces the risk of compromise of the IDS sensors, but does not completely remove it. In April of 2003 a buffer overflow was exposed that would allow an attacker to execute arbitrary code on a Snort system using specially crafted packet fragments.[11] (CERT ca-2003-13) A sensor with a promiscuous interface would be vulnerable to this, as it did not require a direct connection with the device to exploit. While

an attacker could not use this exploit to directly connect to the promiscuous interface, it could try to force the sensor to initiate a connection from a different interface that could be used to interact with the sensor's operating system. This threat vector, coupled with the fact that sensors exploited in this manner could bypass the firewalls, calls for a segregated monitoring subnet. The secondary firewall controls access to and from this subnet and only allows port 25/tcp (SMTP) traffic from the IDS Management server through the secondary and primary firewalls to the Internet for sending alerts to pagers. The only connections allowed into this subnet are from specific security administrator workstations to the IDS Management server on ports 22/tcp (SSH), 80/tcp and 443/tcp (HTTP-HTTPS). The IDS Management server consists of a FreeBSD 5.3 server with a MySQL database for the Snort alerts. Apache 2.0.52 and ACID 0.9.6b23 provide a mechanism for managing and viewing the alerts generated by the sensors. ACID is configured to send email alerts for specified "important" alerts. Snort is also running on the IDS Management server, to monitor the monitoring subnet itself.

Sensors have been placed on every subnet and at strategic traffic gateways. This is to give as complete a picture of what is happening on the network as possible. As the network grows, or more complexity is added in the future more sensors would need to be added to maintain this "big picture."

## Assignment 3

The use of GIAC's website as the point of interaction for many user groups has afforded the opportunity to allow a very small set of ports and protocols through the security gateway devices. The router ingress filter only allows 5 distinct protocols in to the primary firewall. Two of these protocols (500/udp and 50/ip) terminate at that firewall, the other three at devices in the DMZ subnet. The egress filter of the router is more permissive, reflecting the greater degree of trust in the GIAC network. However, rules denying RFC 1918[12] (faqs.org RFC 1918) private addresses beyond the router effectively limit outbound access to anything coming from the primary firewall. This means that the only traffic that will pass outbound is what the firewall policy allows and proxies. This fact will limit the danger that, should GIAC become infected with a virus or worm, it will spread beyond the security gateway, making GIAC a better Internet citizen, or "netizen." The firewall by default denies anything not explicitly allowed, which allows greater confidence that the traffic seen by the router from the GIAC network is expected.

These components by themselves would be insufficient for our security needs. The router by itself would only pass the traffic; it would have no awareness of the relative safety of that traffic. The router would not stop an attack against the reverse proxy over port 80/tcp. The firewall complements the router by providing the protocol intelligence of a proxy. The firewall by itself would be fully exposed to the Internet on all ports. An attack against a vulnerable service running on the firewall would result in compromise. The router shields the firewall ports not absolutely required. In addition, should a misconfiguration or other problem result in internal addresses somehow getting past the firewall, the router serves as a backstop, preventing this from "leaking." This is the essence of defense-in-depth, security components covering the weaknesses of other security components, and providing a safety net in case of failure.

### Router Access Control Lists

Applied to 24.8.16.30 interface (ingress)

| Source | Destination | Ports/Protocols | Action | Description |
|---|---|---|---|---|
| Any | Firewall 24.8.16.32 | 80/tcp (HTTP) | Allow | Allows Internet HTTP access to DMZ |
| Any | Firewall 24.8.16.32 | 443/tcp (HTTPS) | Allow | Allows Internet SSL HTTP access to DMZ |

| | | | | |
|---|---|---|---|---|
| Any | Firewall 24.8.16.32 | 22/tcp (SSH) | Allow | Allows suppliers access to SSH server in DMZ |
| Any | Firewall 24.8.16.32 | 500/udp | Allow | Allows ISAKMP key negotiation to establish VPN tunnel with remote offices |
| Any | Firewall 24.8.16.32 | 50/ip | Allow | Allows IPSEC encapsulation of VPN tunnel from remote offices |
| Any | Any | All | Deny | Blocks everything that hasn't already matched |

The ingress router filter primarily serves to block traffic, with a few exceptions. This reduces load on and protects the firewall. It also greatly reduces the risk potential for the entire network. Only five very specific protocols can pass through the router destined for the firewall. Traffic not of these protocols simply isn't a threat, unless there is a vulnerability in the router.

The order these ACL components occur in is very important. When receiving traffic, this device begins at the top of the list and works down until a match is made. For this reason the final rule is the Deny All. If this rule were at the top of the ACL, our network would be extremely secure, but completely useless. The exceptions to the Deny All appear above this rule, with the Deny All coming in at the end to clean up any leftovers, which will most likely be a large volume of traffic.

Applied to 24.8.16.31 interface (egress)

| Source | Destination | Ports/Protocols | Action | Description |
|---|---|---|---|---|
| 10.0.0.0/8 | Any | Any | Deny | Blocks non-routable address from access to the Internet (RFC 1918) |

| | | | | |
|---|---|---|---|---|
| 172.16.0.0/12 | Any | Any | Deny | Blocks non-routable addresses |
| 192.168.0.0/16 | Any | Any | Deny | Blocks non-routable addresses |
| Any | Any | Any | Allow | Allows all other outbound traffic |

The egress router filter is something of a mirror image of the ingress filter. Where the ingress filter blocked all with some exceptions, our egress filter allows all with a few exceptions. These exceptions are the private address space described in RFC 1918. These addresses by RFC conventions are not routable over the Internet, meaning that they would get blocked at the first Internet router they encountered. But an attacker positioned between the GIAC network and that first Internet router could learn some details about the layout of our network should these addresses "leak" out. In addition, the architecture of our network is such that we should not see these private addresses outside of our firewall. If our router sees these there is a problem somewhere. Letting these addresses out at the least won't help the problem, and may very well contribute to the problem.

Just as above in the case of the ingress filter, order is important to the egress filter. The exceptions to the Allow All final rule are listed before, so that they are the first match should a private address somehow find its way out. Though this Allow All may seem overly permissive, keep in mind that the only traffic allowed through the router is firewall traffic. The firewall rule base must be configured to pass this traffic before it will encounter the router.

**Firewall policy**

Primary firewall

| Source | Destination | Ports/Protocols | Action | Description |
|---|---|---|---|---|
| Any | Reverse proxy 10.2.2.11 | 80/tcp (HTTP) 443/tcp (HTTPS) | Allow | Allows Internet access to reverse proxy |
| Any | SSL VPN 10.2.2.12 | 80/tcp (HTTP) 443/tcp (HTTPS) | Allow | Allows Internet access to SSL VPN |

| | | | | |
|---|---|---|---|---|
| Suppliers | SSH sever 10.2.2.10 | 22/tcp (SSH) | Allow | Allows access from predefined group of supplier servers to GIAC SSH server |
| Security administrator 10.4.4.12 | Router 24.8.16.31 | 23/tcp (Telnet) | Allow | Allows access from Network administrator's workstation to external router |
| Secondary firewall 10.2.2.1 | Any | 80/tcp (HTTP) 443/tcp (HTTPS) | Allow | Allows secondary firewall to proxy internet traffic for internal employees |
| Remote Firewalls | Primary firewall 24.8.16.32 | 500/udp (ISAKMP) 50/ip (ESP) | Allow | Allows predefined group of remote firewalls to establish VPN tunnels with main office |

The firewall policy, or rule base, is mostly concerned with inbound access, but does have some outbound rules. While the ACLs on the router are applied to an interface, the rule base on a proxy firewall is applied "between" the interfaces. The rules can (and should be) specific to inbound and outbound interfaces. The GIAC rule base is fairly simple, only allowing web access, limited SSH access, SSL VPN access, and point-to-point VPN connections. Anything outside these protocols would be stopped at the firewall. Keep in mind it is unlikely the firewall will see traffic outside of these protocols based on the router ingress filters. The firewall will primarily be concerned with governing the allowed protocols in such a way that they are conducted in a safe manner.

The order is not important, as the selected firewall operates on a "best fit" model, not a "first match" like the router. This imposes a little more overhead to process every rule for every connection, but the rule base is small and the total traffic volume will be greatly reduced by the filtering router.

# List of References

"GIAC Certified Firewall Analyst Practical Assignment." GIAC Website Nov. 2004. 12 Dec 2004 < http://www.giac.org/GCFW_assignment.php>

SANS Institute. Track 2 – Firewall, Perimeter Protection, and Virtual Private Networks. Volume 2.4. SANS Press, 2004.

"Cert Advisory CA-2003-13." CERT Advisories Database. Apr. 2003. 12 Dec. 2004. < http://www.cert.org/advisories/CA-2003-13.html>

"Bugtraq ID 4132." Security Focus Vulnerabilities Database. Feb. 2002. 12 Dec. 2004. <http://www.securityfocus.com/bid/4132>

"Bugtraq ID 2936." Security Focus Vulnerabilities Database. Feb. 2002. 12 Dec. 2004. <http://www.securityfocus.com/bid/2936>

"W32.Nimda.gen@MM. " McAfee Virus Information Library. Sep. 2001. 12 Dec. 2004. <http://vil.nai.com/vil/content/v_99209.htm>

"SGS 360R." Symantec Store Appliance Center. 2004. 12 Dec. 2004. http://nct.symantecstore.com/0001/appliance_sgs360R.html

"RFC 1918." Internet RFC/STD/FYI/BCP Archives. Feb. 1996. 12 Dec. 2004.

<http://www.faqs.org/rfcs/rfc1918.html>