Global Information Assurance Certification Paper

GCFW (GIAC Certified Firewall Analyst) Practical Version 2.0

ECONOMICAL PERIMETER SECURITY

This paper discusses a computer network perimeter security proposal for an imaginary company called GIAC Enterprises that is in the business of fortune cookie sayings. The material here begins with some assertions about the company (referred to as GE) and how different users of GE's computer network interact in conducting business operations. An appropriate IP addressing scheme and detailed discussions of the policy configurations of the border router and firewall/VPN appliance including a tutorial on how to configure the border router are also presented. Following this is a section on policy verification planning, execution and evaluation that serves to validate the effectiveness of the proposed design and implementation. This paper concludes with a proposal of how to compromise the perimeter protection of a similar design prepared by an earlier certification candidate. Three separate attacks and targets are discussed.

Jonathan Hosking November 20, 2003

TABLE OF CONTENTS

ASSIGNMENT 1- Security Architecture

GIAC Enterprises (GE) is currently a small, but global company with but one headquarters office. There is a correspondingly small budget for their computer-networking infrastructure. In addition, information technology security tends to command little respect until an unpleasant to catastrophic event occurs that highlights the need for greater resources and attention. Management expects effective security and protection from the evils of the Internet but is also very committed to value, simplicity, and efficiency. Low maintenance & administration are also key objectives in any computer network perimeter design.

Business operations require satisfying six categories of network users:

Assignment 1a- Business Operations/Users of the Network

General Public: will need to connect to GE's web server (www.giacc.com) via HTTP. If and when they wish to become customers, they will be switched to an SSL mode using the HTTPS port and establish an account by specifying a username and password.

Customers (purchasers of online fortunes): will need to connect to GE's public web server initially with HTTP and subsequently with an SSL mode HTTPS session authenticating with a username and password. A secure account access will hopefully be often accompanied by an order! Bulk sayings will be returned such that the customer can specify they be deposited in any file of choice. The sayings are delivered in simple ASCII text resulting in fast transfers and minimal storage requirements. The customer can use any standard word processing program desired to choose a suitable font and subsequently print the sayings. There is a usage agreement similar to typical commercial software agreements that forbid distribution of bulk sayings with or without payment in return.

Suppliers (producers of sayings): will be directed to submit sayings using PGP (or if cost/license is an issue GnuPG version 1.2.2). Documents of sayings will be encrypted, signed, and emailed to GE ensuring authenticity, confidentiality and integrity.

Partners (International translators&resellers): will be able to download sayings with an encrypted HTTPS session authenticating, here too, with a username and password.

Each of the preceding four groups of users are not GE employees but will be allowed appropriate access (web, email, or both). None, as per the company's security policy, will be allowed access to GE's internal network.

GE internal users (employees working in the headquarters office): this group will need Internet http and https for web access, internal email and the ability to correspond with Internet parties. Further, they will require full access to the service network and associated servers.

GE mobile sales and teleworkers (employees not currently in the office): this group of users will be allowed access to internal systems and all access detailed for the internal users group above. By leveraging the security of a VPN, they can use any Internet connection. Many have a cable or DSL connection available in their homes. Also, arrangements have been made with an International ISP to make available local access telephone numbers around the globe for a modest, usage-based hourly access fee. GE insists that all employees be indoctrinated with knowledge of the risks of connecting any systems--especially company issued laptops--directly to the Internet without running a prudently configured personal firewall and virus scanning software.

Here are the traffic flows to be allowed:

Protocol-Port	**Destination**	**Initiating Source=External**
TCP-80	GE's public web server on service network	
TCP-443	GE's public web server on service network	
TCP-25	GE's email proxy running on the firewall	
TCP-76	Remote/mobile user (employee) authentication for VPN usage	
UDP-500 (ISAKMP)	Remote/mobile user (employee) key exchange for VPN use	
ESP	Remote/mobile user (employee) payload encryption for VPN use	
		Initiating Source=Internal
Any-any	Any service network system	
TCP-2784	Firewall's HTTP proxy	
TCP-25	ISP's STMP Mail Relay (from the internal mail server only)	
TCP-77&443	Administration of firewall (encrypted)	
		Initiating Source=Virtual VPN Remote
Any-any	Any service network system	
Any-any	Any Internal	
		Initiating Source=Service
None-none	None	

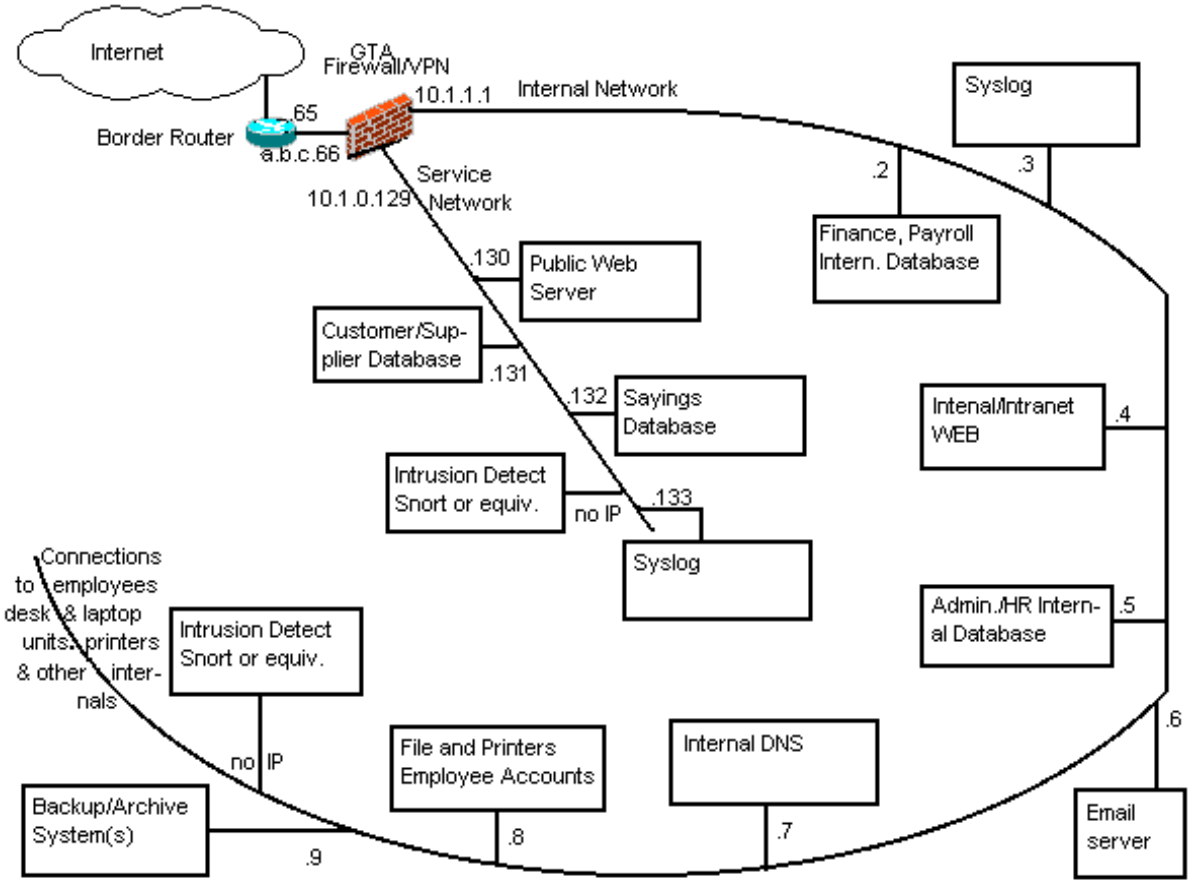
Notice that no system on the service network is allowed to **initiate** a connection with another system **outside of** this segment. GE's arrangement with all parties including customers and partners insists that the "other" will do the initiating and such external interface inbound traffic will be provided for. All packets from GE internal systems and employees will be **initiated** from internal or VPN virtual segment networks. These packets will be allowed full passage to the service network. There will be a great deal of activity here. Sayings will be loaded from

internal servers into the sayings database, customer and partner transactions will be downloaded for billing from the customer/supplier database. Logs will be reviewed, and all servers on the service network checked for intrusions/compromises and backed-up regularly

External DNS (the giacc.com domain) will be managed entirely by GE's ISP. Internal DNS (the giaci.com domain), a small intranet name space, will be managed by a named daemon running on a Linux 7.3 server. Only the Firewall/VPN appliance will be allowed to make Internet DNS lookups to the ISP's cache server. A Firewall/VPN appliance, using a traditional http proxy allows employees to surf the web with reasonable security without allowing any direct Internet access. Occasionally business situations may warrant direct access, and then it would be limited to specific destination ports and IP addresses. The strategy here is to shelter internal systems while at the same time, limit the consequence should these systems ultimately be compromised!

The firewall/VPN appliance offers an email proxy that shields GE's internal email server from unauthorized access through SMTP exploits. It also reduces unsolicited email "spam". The internal email server performs virus scans and neutralizes all inbound and outbound attachments. Outbound email is only allowed to be delivered to an SMTP relay server provided by GE's ISP.

Assignment 1b- Diagram & Components of the Solution



T1 line connecting to an Internet service provider

Four independent IP segments with the following CIDR addresses:

1) Internet: a.b.c.64/26* (a, b and c are substituted for the actual integers)
2) Service network (PSN): 10.1.0.128/25(www.giacc.com natted to external interface as a.b.c.130/25)*
3) Intranet/internal: 10.1.1.0/24*
4) Mobile employees using a VPN virtual segment for access: 10.1.2.0/24

* 100 MBPS switches are used to connect devices on these segments can be of any vendor but the device on the service network must be configurable to have fixed MAC addresses assigned to IP addresses so that ARP poisoning is not effective in allowing a hacker to be able to sniff packets should one system be compromised

Cisco 3620 (sufficient power, proven effectiveness) border router with

1) Access lists/packet filters that only allow in certain source addresses
2) Access lists/packet filters that only allow in certain destination ports

3) Access lists/packet filters that only allow out certain source addresses
4) Access lists/packet filters that only allow out certain destination ports
5) Special functions set to be hardened against recognizance/attacks
6) IOS 12.2
7) Console/configuration only available through physical port

Global Technologies Associates, Inc GB-1500 Firewall/VPN appliance with
1) 3rd separate physical interface for service network (GTA calls PSN)
2) The other two interfaces are internal (GTA calls protected) and external
3) DoS & spoofing attack prevention
4) Hide NAT (internal network IP addresses mapped to F/W's external IP)
5) Stateful packet inspection performed on inbound & outbound packets
6) DHCP server for internal network clients
7) Web Proxy for internal network clients
8) Secure email proxy
9) Remote logging to a separate syslog server
10) IPSec VPN with mobile user authentication

Servers on GE's service network
1) GE's public web server (www.giacc.com)
2) Customer/Supplier Database
3) Sayings Database Syslog server
4) Snort Intrusion Detection

Servers on GE's internal network
1) Internal/intranet web server
2) Syslog server
3) Mail server
4) Administrative & HR Internal Database
5) Internal DNS
6) File, printers, and employee user accounts
7) Backup/archive system(s)
8) Snort Intrusion Detection

Servers are to be Linux-based, run Samba to support Windows-based clients, only listen on necessary ports and be hardened according to the guidelines presented at www.linux-sec.net.

I am suggesting the use of IDSes, Intrusion Detection Systems. These systems are passive devices that are used to monitor all outgoing and incoming network traffic searching for suspicious signatures by using special heuristics and pattern examination. The IDS systems in this design will run on RedHat Linux 7.3 bastion hosts and will each be configured with no IP address. This makes it very difficult for these systems, one of which is placed on the service network and one on the internal network to be compromised or disabled. Each will have Snort Version 2.0 installed watching packets for known attack patterns and logging to the local

machine. These systems will have no other interface and will need to be checked regularly and frequently through the physical monitor and keyboard.

Clients will run Windows 2000 Professional with MacAfee Virus Scan 7. The latter detects and protects against viruses, worms, Java applets and Trojans. The personal firewall aspect of this product protects against various malicious Internet threats. This protection is crucial when mobile clients directly connect to Internet access (from private homes, hotels, etc.) and subsequently use their VPN clients to connect with the GE headquarters internal network.

Assignment 1c- Architecture Discussion

The primary philosophies of this perimeter security design are to block all that isn't necessary to support business operations and, where possible, to defend in depth. The border router employs static filtering and disables unneeded special functions that carry risk. The Firewall/VPN will use rules to statically and statefully filter packets and offers a certified degree of application awareness and inbound and outbound packet inspection as well.

I have selected a Firewall/VPN appliance produced by Global Technologies Associates, Inc to provide both firewall and VPN capabilities. This system is a fully capable firewall built around the concepts of simplicity, power and affordability. The GTA firewall also provides a built-in IPSec VPN. The system is easily managed through any of three interfaces: 1) a simple GUI console interface, 2) a remote SSL web browser client, or 3) a remote proprietary client, GBAdmin, that uses authentication and encryption over tcp port 77. The GTA firewall uses no disk drive. Rather, the security software/operating system is stored in flash memory and the configuration is stored in non-volatile memory. The configuration is compact and is easily uploaded, saved on an administrator's system, downloaded, reloaded and made current, etc.

The International Computer Security Association (ICSA) has certified the GTA firewall family of products to assure that properly configured, internal networks are protected against current threats while allowing important business functions to operate. The certification criteria consist of meeting or exceeding ICSA requirement thresholds in each of the following areas:

Documentation
Logging
Administration
Persistence (if power is lost, then reapplied, is the security policy, logging, authentication, and remote administration intact?)
Functionality (security policy enforcement: passes configured traffic but no other)

Security

- Unauthorized control of administrative functions is blocked
- No vulnerabilities introduced to internal or service network servers
- Not overly vulnerable to Internet community known set of attacks*
  - Invalid packets are dropped
  - Fragmented packet can be dropped as an option
  - ICMP limiting—packets dropped if maximum rate exceeded
  - TCP limiting—packets dropped if maximum rate exceeded

*must not be rendered inoperable by any trivial denial of service attacks and if failure occurs for attacks with no known defense, must fail blocking or closed.

I have always been impressed with the concept of an appliance. In general, appliances dedicated and designed for a sole purpose tend to perform better than more general-purpose systems that can be adapted to perform the task. For example, FAX machines are better than general-purpose systems adapted to the task. In this appliance instance, a network security system and the operating system are an integrated package. The GTA Firewall system is **not** installed on top of an existing operating system with its additional complexity and exposure of other non-security applications. The GTA Firewall system also tends to carry a much lighter burden of bugs, security holes, and patch and release issues than encountered in general purpose operating systems due to its simplicity.

I am recommending the GB-1500 (an enterprise-class device) to provide the required protection and connectivity demands. For the purposes of completing this practical and as a basis for experience and proof of concepts, I have actually configured and tested with GTA's entry-level product in this line, the GB-200 with the VPN option. The configuration process and almost all of the functional capabilities are quite consistent throughout the product line. Features such as hardware VPN acceleration and high-availability characteristics distinguish the higher end products.

ASSIGNMENT 2- Security Policy and Tutorial

Assignment 2a is a narrative of what should be included as part of the border router configuration and why. The complete listing of the Border Router's IOS configuration statements is shown in **Assignment 2b.** Starting with a cleared configuration, the visible IOS statements that Cisco "begins with" appear in black. Any that are superceded by added statements are shown with a ~~strike-through~~. All the added statements and the pertaining portions of the discussion below are shown in one of four colors: Green, inbound access list; Magenta, outbound access list; Blue, global configuration mode; and Red, interface configuration mode. The tutorial presented in **Assignment 2c** details in a step-by-step fashion of how the IOS statements shown in **Assignment 2b** are entered and properly saved into a router's non-volatile memory.

Assignment 2d is a narrative of what should constitute a prudent policy on the Firewall appliance. **Assignment 2e** is the actual GTA firewall's configuration. **Assignment 2f** explains VPN concepts and presents details on how the VPN policy was configured on the GTA firewall. The mobile VPN policy is simply required to match this setting for setting.

Assignment 2a- Border Router Security Policy Discussion

The border router, a Cisco 3620 in this instance, will be configured with policy such that only those addresses, ports, and functionality that is necessary or beneficial to perimeter security is allowed. Further, access to supervision and configuration of the router will be restricted to the console port. This, in turn, will be in a secured room. This filtering strategy will prevent many unnecessary packets from ever reaching the GTA firewall and satisfy the defense in depth objective of this design.

The filtering or screening on the border router acts to limit packets in two main areas: source addresses and destination ports. The restriction of special functions serves to limit attack potential and the unnecessary dissemination of GE's internal network details. Inbound and outbound access lists eliminate unnecessary source addresses and destination ports. Global and interface mode IOS configuration statements are used to disable special functions.

TCP/UDP ports	Designation
1-1,023	Well-known
1,024-49,151	Registered
49,152-65,535	Dynamic and/or Private

Inbound Access List:

(to be applied to incoming traffic on the serial interface)

Source Addresses

The following source IP addresses are to be dropped and logged due to illegitimacy. These could only be the result of an error or deliberate recognizance/attack

Private (including own source): 10.0.0.0/8, 192.168.0.0/16, 172.16.0.0/12
Not allocated & Loopback (127): 0.0.0.0/7, 2.0.0.0/8… 96.0.0.0/3
Not allocated: 173.0.0.0/8, 174.0.0.0/7, 176.0.0.0/5…223.0.0.0/8
Class D, Class E, Multicast, Not allocated & Broadcast: 224.0.0.0/3

Destination Ports

Only packets destined for necessary well-known services/ports are allowed, plus ephemeral/registered & dynamic and/or private "return" ports (tcp & udp port > 1023 & only established if tcp). Ports explicitly allowed: tcp-25 (smtp), tcp-80 (regular web), tcp-443 (SSL web) and these are restricted to the addresses required (firewall's external interface or the service network web server)

Outbound Access List:

(to be applied to incoming traffic on the FastEthernet interface)

Source Addresses

Only let out (our) valid source addresses.... much simpler than the inbound filter a.b.c.66 and a.b.c.130

Destination Ports

Only packets destined for necessary well-known services/ports are allowed, plus ephemeral/registered & dynamic and/or private "return" ports (tcp & udp port > 1023 & only established if tcp). Ports explicitly allowed: tcp-25 (smtp), udp-53 (domain), tcp-80 (regular web), tcp-123 (ntp), tcp-443 (SSL web)

Disabling Special Functions:

IOS statements that are added in global configuration mode are shown in blue below and in Assignments 2b & 2c.
IOS statements that are added in interface configuration mode are shown in red below and in Assignments 2b & 2c.

The first four special functions to be disabled below (representing five configuration statements) are configured by default in Cisco IOS 12.2. They are in effect in the configuration shown in Assignment 2b without needing to enter them explicitly. The remainder, 5 and up, must be added to the configuration.

1. Directed broadcasts can be used to perform a denial of service attacks. "no ip directed-broadcast"

2. ICMP redirects can be used to alter ICMP paths and therefore possibly provide information to a potential attacker. "no ip icmp redirect"

3. TCP and UDP small services include echo, chargen, and discard and are rarely used for legitimate purposes. "no service tcp-small-servers" and "no service udp-small-servers"

4. Finger is a service that is not needed in a properly secured network. "no service finger".

5. IP source routing can be used along with a spoofed address to establish a session with an untrusted host. "no ip source-route"

6. HTTP management access is a wonderful user interface except for the risk imposed. The border router will be managed with the command line interface through the console port. "no ip http server".

7. BOOTP service and the associated exposure will be disabled to prevent denial of service attacks. "no ip bootp server".

8. Service pad allows for alternate management connections. Not needed and an unnecessary exposure. "no service pad"

9. DHCP and the associated exposure are not needed on our border router. "no service dhcp"

10. Prevent the router from sending ICMP unreachables on each interface. This information can possibly be used to learn about our protected networks. "no ip unreachables"

11. Cisco Discovery Protocol (CDP) provides information on all connected Cisco devices. It is unnecessary for this to be known externally. "no cdp enable" on each interface and "no cdp run"

12. Proxy ARP is not needed on our network and can expose us to certain spoofing attacks. "no ip proxy arp"

13. Routers use ICMP redirect messages to notify hosts that a better route is available; this could be used to map internal structure. "no ip redirects"

Though not disabling special functions, the following IOS statements are recommended:

1. Cisco Express Forwarding (CEF) speedups transitions of packets on interfaces and should diminish spoofing possibilities. "ip cef"

2. TCP intercept protects servers from TCP SYN-flooding attacks, which are a type of denial-of-service attack. "ip tcp intercept list aln" Simply define an access list to use in conjunction with the intercept statement. For example: "access-list aln permit ip any any"

3. Keepalives for TCP sessions ensure timely detection of connection failures and also inform routers when sessions are no longer active freeing router resources. "service tcp-keepalives-in" and "service tcp-keepalives-out"

Assignment 2b- Border Router Security Policy

```
version 12.2
no service pad
service tcp-keepalives-in
service tcp-keepalives-out
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
service password-encryption
no service dhcp
!
hostname Border_Router
!
ip subnet-zero
ip tcp intercept list 103

no ip source-route
ip cef
!
no ip bootp server
!
interface FastEthernet0/0
 no ip address
 ip address a.b.c.65 255.255.255.192
 ip access-group 102 in
 no ip redirects
 no ip proxy-arp
 no ip unreachables
 no cdp enable
 shutdown
 half-duplex
!
interface Serial0/0
 service-module t1 timeslots 1-24
no ip address
```

```
encapsulation frame-relay IETF
frame-relay lmi-type ansi
 shutdown

interface Serial0/0.1 point-to-point
frame-relay interface-dlci 500 IETF
ip access-group 101 in
no ip redirects
no ip proxy-arp
no ip unreachables
no cdp enable
! w, x, y, and z substitute for the actual integers
ip address w.x.y.z 255.255.255.252

!
ip classless
ip http server
no ip http server
ip pim bidir-enable
!
! INBOUND ACCESS LIST
access-list 101 remark block “not allocated” source addresses
access-list 101 deny ip 0.0.0.0 1.255.255.255 any log
access-list 101 deny ip 2.0.0.0 0.255.255.255 any log
access-list 101 deny ip 5.0.0.0 0.255.255.255 any log
access-list 101 deny ip 7.0.0.0 0.255.255.255 any log
access-list 101 deny ip 23.0.0.0 0.255.255.255 any log
access-list 101 deny ip 27.0.0.0 0.255.255.255 any log
access-list 101 deny ip 31.0.0.0 0.255.255.255 any log
access-list 101 deny ip 36.0.0.0 1.255.255.255 any log
access-list 101 deny ip 37.0.0.0 0.255.255.255 any log
access-list 101 deny ip 39.0.0.0 0.255.255.255 any log
access-list 101 deny ip 41.0.0.0 0.255.255.255 any log
access-list 101 deny ip 42.0.0.0 0.255.255.255 any log
access-list 101 deny ip 49.0.0.0 0.255.255.255 any log
access-list 101 deny ip 50.0.0.0 0.255.255.255 any log
access-list 101 deny ip 58.0.0.0 1.255.255.255 any log
access-list 101 deny ip 70.0.0.0 1.255.255.255 any log
access-list 101 deny ip 72.0.0.0 7.255.255.255 any log
access-list 101 deny ip 83.0.0.0 0.255.255.255 any log
access-list 101 deny ip 84.0.0.0 3.255.255.255 any log
access-list 101 deny ip 88.0.0.0 7.255.255.255 any log
access-list 101 deny ip 96.0.0.0 31.255.255.255 any log
access-list 101 deny ip 173.0.0.0 0.255.255.255 any log
access-list 101 deny ip 174.0.0.0 1.255.255.255 any log
access-list 101 deny ip 176.0.0.0 7.255.255.255 any log
```

```
access-list 101 deny ip 184.0.0.0 3.255.255.255 any log
access-list 101 deny ip 189.0.0.0 0.255.255.255 any log
access-list 101 deny ip 190.0.0.0 0.255.255.255 any log
access-list 101 deny ip 223.0.0.0 0.255.255.255 any log
access-list 101 remark block class D&E, multicast, not allocated & broadcasts
access-list 101 deny ip 224.0.0.0 31.255.255.255 any log
access-list 101 remark block private source addresses
access-list 101 deny ip 10.0.0.0 0.255.255.255 any log
access-list 101 deny ip 192.168.0.0 0.0.255.255 any log
access-list 101 deny ip 172.16.0 0.15.255.255 any log
access-list 101 remark block our addresses from coming in
access-list 101 deny ip a.b.c.64 0.0.0.63 any log
access-list 101 deny ip a.b.c.128 0.0.0.127 any log
access-list 101 remark keep out slammer worm/ VERY PREVALENT
access-list 101 deny udp any any eq 1434
access-list 101 remark allow web traffic to www.giac.com
access-list 101 permit tcp any host a.b.c.130 eq www
access-list 101 permit tcp any host a.b.c.130 eq 443
access-list 101 remark allow incoming mail
access-list 101 permit tcp any host a.b.c.66 eq smtp
access-list 101 remark allow incoming authentication for VPN purposes
access-list 101 permit tcp any host a.b.c.66 eq 76
access-list 101 remark allow esp for VPN
access-list 101 permit esp any host a.b.c.66
access-list 101 remark allow isakmp for VPN
access-list 101 permit udp any host a.b.c.66 eq isakmp
access-list 101 remark allow return/ephemeral ports to be opened
access-list 101 permit tcp any host a.b.c.66 gt 1023 established
access-list 101 remark allow Network Time packets
access-list 101 permit udp any host a.b.c.66 eq ntp
access-list 101 remark block and log everything else
access-list 101 deny ip any any log
!
! OUTBOUND ACCESS LIST
access-list 102 remark allow Network Time packets
access-list 102 permit udp host a.b.c.66 any eq ntp
access-list 102 remark allow isakmp for VPN
access-list 102 permit udp host a.b.c.66 any eq isakmp
access-list 102 remark allow internal mail to go to ISP's mail relay
access-list 102 permit tcp host a.b.c.66 mail.relay.ip.addr eq smtp
access-list 102 remark allow http proxy on f/w to the web
access-list 102 permit tcp host a.b.c.66 any eq www
access-list 102 remark allow proxy on https to the web as well
access-list 102 permit tcp host a.b.c.66 any eq 443
access-list 102 remark allow esp for VPN
access-list 102 permit esp host a.b.c.66 any
```

```
access-list 102 remark allow return/ephemeral ports to be opened
access-list 102 permit tcp host a.b.c.66 any gt 1023 established
access-list 102 remark allow access for Internet DNS
access-list 102 permit udp host a.b.c.66 host isp.dns.cache.server eq domain
access-list 102 remark allow return/ephemeral ports to be opened
access-list 102 permit tcp host a.b.c.130 any gt 1023 established
access-list 102 remark deny ip any any log
access-list 102 deny ip any any log

! TCP INTERCEPT ACCESS LIST
access-list 103 permit ip any host a.b.c.66
access-list 103 permit ip any host a.b.c.130

no cdp run
!
line con 0
exec-timeout 5 0
password 7 A538D7647FA0
line aux 0
line vty 0 4
no login
no exec
transport input none
!
banner motd ^c
Unauthorized use is prohibited; violators may be prosecuted!
^c
end
```

Assignment 2c- Tutorial on Cisco 3620 Border Router

Configuration of Cisco routers is performed via a command line interface through an ASCII terminal (actual or emulated through a PC's serial port). Cisco supplies the necessary cables and connectors with all of their equipment. The communication port settings are 9600 baud, 8 data bits, no parity, and one stop bit. Pressing the Enter/Return key begins a session and must also conclude any command/statement argument sequences that follow in this discussion. Successfully satisfying the prompt for a password (there are additional authentication modes that have the benefit of logging any configuration changes made by this user) will place the user into what is termed user-level executive mode. This mode can be recognized by a prompt consisting of the router name followed by an angle bracket (>). Here, certain basic commands and basic router information is available. To get to the privileged executive mode where more detail about the routers configuration is available and more consequential commands can be submitted, the user types "enable" at the router prompt. An additional password may be required. This mode can be recognized by a prompt consisting of the router name followed by a pound sign (#).

When entering commands or keywords, you can abbreviate them to the fewest number of characters that make them unique. For example, the command "configure terminal" can be entered by just typing the characters "conf t" noting that the space between the "f" and the "t" is imperative as there is no IOS command that begins with "conft". Some commands will result in more than one screenful of response and the text "—More—" appears near the bottom of the screen. Press the spacebar to be shown the next screenful, press Enter/Return to be shown just the next line, or any other key to return to the prompt.

You can obtain context-sensitive guidance through the use of the question mark (?). Use this at a prompt for a list of all the commands that could apply. For all commands that begin with a certain sequence of characters, enter just as many characters that your sure of immediately followed by the "?". Finally a "?" can be used at any position in the line to be shown all possible keywords or arguments that could apply.

In order to add or delete IOS statements that effect the configuration of the router, it is necessary to enter yet another mode from the privileged-level executive mode. Typing "configure terminal" or the abbreviation "conf t" initiates global configuration mode. The prompt here now changes to include "(config)" between the router name and the pound sign. Commands entered here apply to the router as a whole. These are shown in Assignment 2a (router policy discussion) and Assignment 2b (actual configuration) in blue (and green and magenta as the access-list definition commands are really global configuration mode items as well). The final mode discussed in this tutorial is the interface configuration mode. If not yet clear, each mode is incremental and necessitates that you enter the prior mode before proceeding to the next. Any given IOS

statement is either valid or not in any given mode. To return to the prior mode, use the IOS command “exit”. In general, to add configuration statements, just type them followed by Enter/Return. To delete such a statement just type it preceded by the three characters “no “. Now, let’s return to our discussion of “interface configuration mode”. To enter this mode from “global configuration mode” use the IOS “Interface” or “Int” statement followed by the name of any interface (ex. FastEthernet0, Serial 0/0, etc.). The “config” portion of the prompt presented in global changes to be “config-if” such that the entire prompt would now appear as “Router-Name (config-if)#”. The statements shown in the color red in Assignment 2a (router policy discussion) and Assignment 2b (actual configuration) are entered in the “interface configuration mode”.

A few of the more useful key sequences (valid in all of the modes) are:

Ctrl-P to recall the last line typed.
Ctrl-B to move the cursor backwards
Ctrl-F to move the cursor forward
Delete or Backspace to erase characters
Ctrl-U to delete the entire line you have been typing

The following table summarizes the four different modes previously discussed:

Mode	What is entered	Enter from prior mode by typing
User-level executive	certain basic commands	(user)/password
Privileged-level executive	all commands	enable
Global configuration	configuration statements & ALs	configure terminal
Interface configuration	configuration statements	any interface name

Access Lists

The access lists described here are known as Cisco Extended Access-Lists and provide the capability to drop certain packets based on IP address and port number characteristics. Access lists are defined in global configuration mode and consist of a collection of statements all beginning with access-list nnn, where nnn is a number 100-199. These statements are kept and processed in the order entered. Additional statements can be appended at any time but will invariably be added (after) any prior statements. In order to change any statement entered previously or insert a statement in any other sequence, it is necessary to delete the entire access-list and reenter all the statements desired in the exact order. Cut/copy/paste functionality of any convenient text editor in conjunction with your terminal emulation program is useful for such modifications. All statements representing any given access-list can be deleted by simply typing “no access-list nnn”. Access-lists can be applied to any given interface under the interface

configuration mode for that interface by entering "ip access-group nnn in" or "ip access-group nnn out". In or out represent the direction of packet travel and there can be only one list so applied "in" and one so applied "out" for each interface. As an example, if we were to apply an access-list in on the serial interface that connects to our Internet service provider, it would operate on packets coming from Internet sources destined for our systems. When an access list is written, there is an implicit "deny all" at the end of the list. If an undefined access list is applied to an interface, this has the effect of permitting all packet traffic. There are two general ways to compose access lists; 1) allow only what traffic you desire to pass and then let the remainder be blocked, or 2) first deny that traffic you don't want and then let the remainder pass. These methods can be combined to achieve certain special objectives. For instance, if you want to determine how much of a certain type of traffic is present, simply deny that traffic and log the instances of the rejected packets.

Extended Access Lists

Extended access-list syntax (the usual conventions for |, { } and [ ]apply):

access-list *aln* {deny | permit} *protocol source destination* [*operator port/type*] [log] [established]

aln: access-list number=100-199
protocol: {ip | tcp | udp | icmp | ah | esp | 0-255} (ip includes tcp, udp, & icmp)
source: {host *ip-address* | any | *source-network wildcard-mask*}
destination: {host *ip-address* | any | *destination-network wildcard-mask*}
A standard dotted decimal notation represents a host or group of hosts for the source-network and destination-network fields. The wildcard-mask functions in a reverse manner to a subnet mask. That is, binary zeroes insist on a positional match and binary ones "don't care".
operator: {eq | lt | gt }
port/type: {name or number of udp or tcp port | name or number of icmp-type}
example for tcp might be smtp or 25, example for icmp might be echo or 8

The "established" keyword allows packets only if the ACK or RST bits are set (TCP only). We take advantage of this to allow connections to be established using necessary ephemeral "return" ports that are desired and to disallow normal unsolicited traffic destined to GE's systems referencing destination TCP ports > 1023. At first, I was very excited by this keyword and how it would enhance my policy. Internet initiated connections seeking this wide range of destination ports would be stopped cold at the border router. Defense in depth achieved indeed. However, I soon realized that such would apply only to those packets obeying the rules! The truly evil intentioned probes and packets should certainly be expected to craft packets fiddling with ACK and RST bit in order to get past border router policy.

To apply an Access List to an interface:
ip access-group *aln* {in | out}

After completing the adding, deleting and modifying of IOS configuration commands, you'll want to type "exit" (and perhaps a second "exit", if you were in interface configuration). Notice that "config" is now absent from the prompt but the "#" indicates privileged executive mode.

Type "show running" to have the current configuration statements listed. You may return the configuration modes by typing "conf t" again. When the configuration is as desired, it can be saved for future router restarts, power failures, etc. by typing "write memory". The IOS command to cause the router to restart is "reload". Here the router's running configuration will be loaded with the last version of IOS configuration statements written to non-volatile memory.

Assignment 2d- Firewall Security Policy Discussion

The Firewall/VPN appliance policy discussion describes the GTA Firewall Configuration Summary shown in Assignment 2e. Four discontiguous VPN pertaining passages are shown in a red color and discussed in Assignment 2f while the non-VPN aspects are shown in black and discussed beginning on the next page. All the GTA Firewall appliances have at least three separate interfaces. Three are used in the design here. GTA names these as follows:

External	Connects thru the Border Router's external interface to Internet
Protected	Connects to the internal segment (referred as internal in my writing)
PSN	Connects to service network (referred to as service in my writing)

PSN stands for Private Service Network and the design strategy is to allow external systems (those on the Internet) to initiate connections with specific systems, protocols and ports on this segment and allow responses. As configured to this point, we are providing access only to GE's web server, www.giac.com, tcp ports 80 and 443 only. Remote VPN connected employees and all internal systems will be allowed full access (all addresses, protocols and ports) of service network systems. However, no service segment system will be allowed to initiate any connection with any system, port or protocol on any other segment.

Network Address Translation (NAT), active by default on the GTA firewall, translates IP addresses sourced on either the internal or service segments as they pass through the firewall to either the service or external segments. Addresses will be translated to the firewall's interface address on either of these segments. Although such behavior is generally useful, there are two instances in which we need to override this in order to institute the desired policy.

In the first instance, we need to allow public, external access to the www.giacc.com web server, which is connected physically to the service network. From a public and external prospective, www.giac.com is at a different IP address (and even a different IP address segment) than GE's GTA firewall's external address (giacc.com). The following three provisions configure a special instance of NAT:

1) **Aliases:** that the external interface is to represent an alias name of www.giacc.com at ip address a.b.c.130.

2) **Inbound Tunnels:** packets coming to the external interface destined for a.b.c.130 are to be directed to the service network system at IP address 10.1.0.130. (two tunnels: http (port 80) and https(port 443))

3) **Static Address Mappings:** an internal or service network address is mapped to an alias name (typically associated with the external interface). In this case, 10.1.0.130 is mapped to www.giacc.com

In the second case of needing to override default NAT, we would like no NAT at all to take place on those packets originating on the internal network bound for the service network. IP Pass Through is the GTA Firewall's solution that selectively disables NAT. Here, both a "Filters" and a "Hosts/Networks" declaration specify the IP address ranges, interfaces and direction where addresses remain unchanged.

Refer now to Assignment 2e--starting at the top. **Basic Configuration** contains functions that address basic GTA firewall setup and configuration. Two Internet DNS cache servers are defined in order to allow the firewall to make DNS lookups. The cache servers are made available to us by the Internet Service Provider (ISP) and also define any of GE's names, such as www.giacc.com and giacc.com, which are necessary to conduct business on the Web. Our primary Internet domain is specified as giacc.com. Internal network systems and service network systems don't require and hence are not allowed to access these external DNS cache servers. Features include the software release version and extras that have been optionally activated—such as VPN. Network Information defines the interfaces, IP addresses in CIDR form, MAC addresses, default gateway and hostname. The default gateway references the border router's IP address. The hostname is really just a name assigned to the firewall. This name is used and found in the logs. Preferences simply details administrative contact information.

Next is the **Services** section. DHCP server is enabled in our configuration for protected network clients. DHCP automates the process of assigning IP addresses and all the information shown is employed in this process. For DNS, an internal name server is used and this is completely isolated from Internet DNS. The Email Proxy facility is also enabled in our configuration and is used to

shield the internal mail server (on the internal network) from unauthorized access and to reduce spam. Senders are validated against mail abuse prevention RBLs (Real-time Blackhole Lists). Matches are dropped and a "do not send again" packet is returned to the source. An outbound filter is defined to allow the Internal mail server to send all outbound messages to GE ISP's SMTP relay server. GE's internal mail server includes software to scan and clean any viruses and worms encountered in all inbound and outbound attachments. The Network Time facility is enabled to synchronize the GTA firewall with Internet time service authorities. Remote logging is directed to the internal network syslog server. Records of voluminous activity can be retained for reference. Site personnel should review these logs regularly. There are several packages available that can prescreen logs so that human eyes can be saved for the most remarkable entries! Many situations arise where consulting past logs proves invaluable. The SNMP service, a facility for managing IP devices and retrieving a wide range of networking metrics, is to remain disabled in this configuration. It seems too great a risk of dissemination of network information to parties that shouldn't have it.

The **Authorization** section follows the Services section. The first element is the Administration accounts and the administration access options: Web and/or Remote Management Console: GBAdmin (a proprietary management client). The final Authorization section element in our configuration is Users definitions as pertain to the Mobile VPN and will be discussed in ASSIGNMENT 2f- VPN Security Policy Discussion.

The **Content Filtering** section pertains to web site access and control for internal users (even when connecting through the VPN). We have configured a traditional http proxy using port 2784. The proxy allows for better protection of internal network systems, as there is always an intermediate or buffering system (in this case the firewall itself) between a GE internal system and whatever one ends up communicating with on the Internet! It should be noted that this choice does consume some measurable processing resources on the GTA Firewall. However, with a transparent http proxy or just an outbound filter, packets would simply be forwarded unexamined. We have not, at this time, chosen to limit the sites that can be accessed.

The Routing section provides three facilities for special routing techniques. We don't make use of any of these in this design.

The **Objects** section allows the definition and naming of certain special IP addresses or segments in CIDR format. Two definitions were declared: 1) ANY_IP = 0.0.0.0/0 and 2) External Interface = a.b.c.66. The other element in this section is VPN Objects and will be discussed in ASSIGNMENT 2f- VPN Security Policy Discussion.

Next is the **Filters** section. Filters control access to and through the GTA firewall. Outbound and Remote Access filters are defined in the Filters section.

Time Groups (which can specify the times a filter applies), a set of protocol to name associations, and several "preferences" are also found in this section. Pass Through filters, which use the same mechanisms and syntax for filter management as the Outbound and Remote Access ones, are defined in the IP Pass Through section.

A filter set is all the filters for a specific type. The order of the set is important. Each packet is compared to the appropriate set (Remote Access, Outbound, or IP Pass Through) from top to bottom. For each filter, one of three conditions will prevail:

	condition	resulting action
1)	no match	check next filter, if set exhausted, packet is rejected
2)	match accept	packet accepted
3)	match deny	packet rejected

Outbound filters apply to packets sourced on the internal and service networks that are destined for the Internet (which is out the external interface, to the border router and beyond).

Remote Access filters operate on inbound packets (those created by external sources bound for either the internal or service networks.

IP Pass Through Filters control access to a certain host or networks that have been defined as IP Pass Through addresses and for which network address translation (NAT) is suspended. These filters are different than the preceding types in that they control inbound or outbound access to or from the designated IP Pass Through hosts/networks or VPN virtual networks.

Certain automatic filters are generated by the firewall (on the fly) to accommodate the arrival of expected response packets. Although automatic filters can be disabled, this generally defeats the secured access the firewall is employed to provide. Automatic filters can also be logged (typically for troubleshooting purposes).

FILTER PREFERENCES

Filter Preferences allows the setting of many logging and filtering options available on the GTA firewall.

Alarms

This section allows the parameters for alarm notifications to be set. When a filter (Remote Access, Outbound, or IP Pass Through) is matched, an alarm event is activated. Each alarm event increments the alarm counts by one. If either the time or number of alarms threshold is exceeded, a notification will be sent

documenting all the events that were contributors. Multiple messages will be sent if the number of events exceeds the maximum alarm count.

Email Server
Enable this option and specify an Email server's IP address and user name in order to receive notifications.

General
All of the following seven aspects can be logged or not. Some, unless noted, can be enabled or disabled. Additional control options are detailed in each description.

1) Automatic Filters (described previously)

2) Deny Address Spoof (always enabled)
A spoof occurs when a packet arrives at one interface and its return path is through a different interface. This may be caused by an intrusion attempt (altering the packet source IP address); or may be due to a misconfigured firewall. Additional options include generate an alarm event or email.

3) Deny doorknob twist (always enabled)
A doorknob twist occurs when a connection is attempted on a port for which there is no service or tunnel in place and a filter has accepted the packet. A Doorknob Twist usually indicates that the firewall is misconfigured. Additional options include generate an alarm event, email, or generate an ICMP "service not available" message to send to the source IP address of the attempted connection.

4) Deny fragmented packets
Unless enabled, fragmented packets are reassembled and forwarded as long as permitted by the security policy. If necessary, this preference can be enabled to thwart fragment attacks.

5) Deny invalid packets (always enabled)
If a packet is not the expected size or has an invalid option bit, the firewall denies the packet. For example, an ICMP port unreachable packet must have at least 28 data bytes.

6) Deny unexpected packets (Always enabled)
If a packet is valid, but not expected by the state table, the firewall denies it. A packet can only generate a single ICMP port unreachable response; a second one may indicate an ICMP replay attack. Also, an unexpected packet may be a packet that does not have the correct flags during TCP's three-way handshake.

7) Stealth Mode

In stealth mode, the GTA Firewall will not respond to ICMP packets such as ping or traceroute. In addition, UDP traceroutes are not functional.

Default Logging

Every filter has a log action. A "Yes" in the filter action field for the filter explicitly logs the packet. A "No" explicitly does not log it. The Default option specifies the action defined here. By default, all rejected packets for all protocols are logged. Tunnels refer to connections created by the action of a filter (automatic or user-defined) or an inbound tunnel. The following three default aspects are always enabled. For each, there is the option to log or not.

1) Filter Blocks
2) Tunnel Opens
3) Tunnel Closes

Pager

Connect a modem to an available serial port on your GTA Firewall to send a message code to a pager.

SNMP

Simple Network Management Protocol (SNMP) is a standard for managing IP devices, retrieving data from each device on a network, and sending it to designated hosts. If SNMP is checked as an action AND SNMP was enabled in the Services section (which it is not), the GTA Firewall will generate an enterprise-specific generic trap on a filter definition.

Fields that make up a filter:

1) Description: any text that helps to identify the filter's purpose
2) Disable: check to disable the filter
3) Type: accept or deny the packet's passage
4) Interface: physical interface on which the packet to be filtered filter will arrive. <ANY> will "filter" on all interfaces.
5) Protocol: TCP, UDP, ICMP, IGMP, ESP, AH, ALL, or any other protocol defined in the Protocols section can be selected to match against the packet.
6) Priority: A notice sent with the alarm event. Defined by the user.
7) Authentication Required: Check to require that users allowed by this filter authenticate to the firewall using the GBAuth utility.
8) Actions: Select one or more events to notify the administrator about a filter event. Alarm, Email, ICMP, Pager, SNMP, Stop Interface.
9) Log: Yes, No, and Default. "Yes" logs all events for this filter, including accepts. "Default" logs the filter event as defined Filter Preferences section. "No" does not log any filter events for this filter. Selecting "yes" will create many log events, and so is used mostly for configuration testing.
10) Time based: Click to make the filter operate at a specified time.
11) Time group: Select the previously created time parameters from the dropdown box.

12) Source Address: IP address of the packet. The selected IP address or object will be matched against the source IP address of the packet.
13) Range: select to choose a range of ports.
14) Source Ports: Leave all zeroes for any source port to be accepted. The source port can be a single port, multiple ports or a range of ports. Specified Source Ports are matched against the source port of the IP packet. Destination Address: IP address or object will be matched against this field of the packet.
15) Range: select to choose a range of ports.
16) Broadcast: select Broadcast if this is a Broadcast Destination.
17) Destination Ports: sometimes called services. Same aspects as Source Ports

Actual filters defined:
Outbound:
#1: Allow outbound packets from internal mail server to our ISP's SMTP relay

Remote Access:
#1: Allow inbound access on tcp ports 80 and 443 our web server (www.giac.com)
#2: Allow network time protocol packets to be returned from Internet-based NTP servers.
#3: Allow external system access to the GTA firewall's authentication server (GBAuth on TCP port 76). Traffic flows are encrypted.

Remote access rules #4 and #5 pertain to VPNs and are covered ASSIGNMENT 2f- VPN Security Policy Discussion.

#6: Allowed protected (internal) network access to the GTA firewall on TCP Ports 77 and 443. These are encrypted connects for administrating the firewall.
#7: Allow protected (internal) and VPN users access to the http proxy.
#8: Allow Internet mail servers to be able to send to the email proxy.
#9: Block and log all other protocol and port access attempts.

The **IP Pass Through** Section follows the Filters Section. Filter #1 allows the internal network to initiate any access desired to service network systems. Pass Through Filter #2 pertains to VPN aspects and will therefore also be covered ASSIGNMENT 2f- VPN Security Policy Discussion. The second aspect of this section, the Hosts/Networks definition, is used to specify an IP address or network that will not have NAT applied to the packets. In our configuration, internal network addresses shouldn't be translated when they are passed to the service network.

The last section in our GTA firewall policy is **NAT** (Network Address Translation). One of the values of NAT is to protect knowledge of the internal network's IP addressing scheme from non-GE employees. The NAT facility is active by default and is applied (unless otherwise configured as pass through) to all

outbound packets coming into either the service or external networks. The default NAT is also referred to as dynamic. The other form of NAT available on GTA firewalls is Static NAT**.** Static NAT is used in our policy to translate the actual IP address of the www.giacc.com web server into the publicly defined and referenced one. There are four sub-sections to the NAT section of our GTA firewall policy: Aliases, Inbound Tunnels, Static Address Mappings, and Timeouts.

The Alias facility allows a network interface to be represented by multiple IP addresses. An IP alias can be assigned to any interface. In our configuration, the alias name www.giacc.com is assigned to the external interface and associated with ip address a.b.c.130. Two inbound tunnels are defined in our configuration; one for port 80 (http) and one for port 443 (https/SSL). The tunnels associate the a.b.c.130 public (www.giacc.com) address with the actual service network (PSN) address of 10.1.0.130. The final necessary definition binds everything together in an explicit Static Address Mapping (also known as the outbound mapping) definition. This requires an internal "from" ip address or network and an alias definition as a "map to". Of course, in our policy, the from is 10.1.0.130 and the "map to" is a.b.c.130.

Timeout settings round out the discussion of the last sub-section in the last section (NAT) of our GTA firewall security policy. Timeouts define how long a connection should be idle before it is considered a candidate for closure. Protocols TCP, UDP, and ICMP each have their respective values and the "Default" metric will apply to any other protocol. TCP, due to its connection-oriented operation, has additional settings. Keepalives are recommended and enabled in our configuration (just as they were specified on the border router) as they allow inactive and questionable connections to be cleaned-up more expeditiously thereby reducing resource demands and risk exposure at the same time.

Assignment 2e- Firewall/VPN Security Policy

GNAT Box Software Configuration Summary

GB-200 Version: 3.4.0 Mon 2003-11-10 10:04:42

Basic Configuration

DNS

External name server: 198.6.1.4 198.6.1.122
Internal name server:
Domain: giacc.com
DNS Proxy: Disabled

Features

GB-200 3.4 - Registered (10 users)
GB-200 3.4 - VPN

Network Information

LOGICAL INTERFACES

Name	Type	IP Address	NIC	DHCP
EXTERNAL	EXTERNAL	a.b.c.66/26	sis1	
PROTECTED	PROTECTED	10.1.1.1/24	sis0	
PSN	PSN	10.1.0.129/25	sis2	

NETWORK INTERFACE CARDS

NIC	MAC Address	MTU	State	Connection
sis0	00:01:24:C7:D6:00	1500	up	AUTO
sis1	00:01:24:C7:D6:01	1500	up	AUTO
sis2	00:01:24:C7:D6:02	1500	up	AUTO

Default gateway: a.b.c.65
Hostname: giacc.com

Preferences

ADMINISTRATOR CONTACT INFORMATION
Name: Jonathan Hosking
Company: GIAC Enterprises
Email Address: jhosking@giacc.com
Phone number: 555 555-4126

Support email address: netadmin@giacc.com
Default character set: ISO-8859-1

Services

DHCP Server

1 #Serve addresses to protected (internal) users
Beginning: 10.1.1.100
Ending: 10.1.1.199
Mask: 255.255.255.0
Lease time: 1440 minutes
Domain: giaci.com
Name server: 10.1.1.7
Gateway: 10.1.1.1

Email Proxy

Enabled: yes
Primary server: 10.1.1.6
Alternate server:
Time out: 120 seconds
Maximum connections: 50
Domain: giacc.com
Use MX: yes
Verify RDNS: no
Maximum size: 5000 kilobytes
MAPS 1: enabled relays.ordb.org
MAPS 2: enabled list.dsbl.org
MAPS 3: disabled blackholes.mail-abuse.org
MAPS 4: disabled relays.mail-abuse.org

Network Time Service

	Server	key
1	129.6.15.28	0
2	132.163.4.102	0
3	129.6.15.29	0

Remote Logging

Logging system messages to server: 10.1.1.3:514
Filter facility: local1
NAT facility: local0
WWW facility: local2

SNMP

disabled

Authorization

Admin Accounts

Lockout: enabled
Threshold: 5
Duration: 300
Notification: enabled

Index	User	Permissions
1	gtaigiac	admin console www remote

Remote Admin/Authentication

WWW server: enabled
Updates: enabled

Port: 443
Encryption: high

RMC server: enabled
Updates: enabled
Port: 77
Encryption: high

AUTH server: enabled
Updates: enabled
Port: 76
Encryption: high

Users

1 Sample
Description: verify functionality of mobile VPN for one user
Identity: jon@giacc.com
Auth method: password
VPN object: MOBILE
Remote network: 10.1.2.1
Security associations: 2

Content Filtering

Access Control Lists

1 #
Source address: ANY_IP
Filtering facilities: LocalDenyList

MOBILE CODE BLOCKING
JAVA blocking: disabled
JAVA script blocking: disabled
ActiveX blocking: disabled

Local Content Lists

Allowed
Not configured

Denied
Not configured

Preferences

Transparent: disabled
Traditional: enabled
Proxy port: 2784
Block action: Use message
Message: Local policy denies access to web page.
URL:

Routing

Gateway Selector

disabled

RIP
disabled

Static Routes
Index	IP Address	Gateway

Objects
Addresses
1 ANY_IP - DEFAULT: Matches all IP addresses.

Index	Member
1	0.0.0.0/0

2 External Interface - External Interface f/w = giacc.com

Index	Member
1	a.b.c.66

VPN Objects
1 #GIAC Ent.: MOBILE VPNs
Name: MOBILE
Authentication required: yes
Gateway: EXTERNAL
Force mobile protocol: no
Local network: 10.1.0.0/23
Phase 1: aggressive 3des hmac-sha1 group 2
Phase 2: 3des hmac-md5 group 2

Filters
Outbound
1 # Allow outbound email access to ISP's SMTP relay server
Accept notice "PROTECTED" TCP
from 10.1.1.6
to 199.171.54.245 25

Remote Access
1 # Allow inbound access to www.giac.com
Accept notice "EXTERNAL" TCP
from ANY_IP
to a.b.c.130 80 443

2 # Allow network time protocol call returns to f/w
Accept notice ANY UDP
from ANY_IP 123
to External Interface 123

3 # Allow access to user authentication service, GBAuth on f/w (TCP 76)
Accept notice "EXTERNAL" TCP
from ANY_IP
to External Interface 76

4 # Allow key exchange with f/w for VPN
Accept notice "EXTERNAL" UDP
from ANY_IP 500
to External Interface 500

5 # Allow ESP to f/w for VPN encrypted packets
Accept notice "EXTERNAL" 50
from ANY_IP
to External Interface

6 # Allow protected network access to admin services.
Accept notice "PROTECTED" TCP
from 10.1.1.0/24
to 10.1.1.1 443 77

7 # Allow protected & VPN users to use traditional WWW proxy
Accept notice ANY TCP
from 10.1.0.0/22
to 10.1.1.1 2784

8 # Allow Internet mail servers connections to email proxy
Accept notice ANY TCP
from ANY_IP
to External Interface 25

9 # DEFAULT: Block with notice any other access to all interfaces.
Deny warning ANY ALL
from ANY_IP
to ANY_IP

Time Groups

Not configured

Protocols

Index	Name	Number
1	IGMP	2
2	GRE	47
3	ESP	50
4	AH	51

Preferences

GENERAL

	Enable	Alarm	Email	ICMP	Log
Automatic filters	X				
Deny address spoof	X	X			X
Deny doorknob twist	X	X			X
Deny fragmented packets					
Deny invalid packets	X				
Deny unexpected packets	X				
Stealth mode	X				

DEFAULT LOGGING

Tunnel opens	X				
Tunnel closes	X				X
Filter blocks	X				X

ALARMS
Send email for alarms when 10 seen within 120 seconds.
Send a maximum of 500 alarms per email.
Do not attempt to log host names using reverse DNS.
Do not attempt to send page when alarm threshold reached.

EMAIL SERVER
Server name: giacc.com
From:
To: netadmin

SNMP TRAPS
disabled

PAGER
disabled

IP Pass Through

Filters

1 # Accept all traffic from protected network to PSN
Accept notice "PROTECTED" ALL
from 10.1.1.0/24
to 10.1.0.128/25

2 # Allow inbound access from mobile clients
Accept notice "EXTERNAL" ALL
from 10.1.2.0/24
to 10.1.0.0/23

Hosts/Networks

Index	From	Interface	Options
1	10.1.1.0/24	PSN	outbound

NAT

Aliases

Index	Interface	Name	IP Address
1	EXTERNAL	www.giacc.com	a.b.c.130/25

Inbound Tunnels

1 # www.giacc.com http
TCP from www.giacc.com:80 to 10.1.0.130:80
2 # www.giacc.com https
TCP from www.giacc.com:443 to 10.1.0.130:443

Static Address Mappings

Index	From	To IP Address
1	10.1.0.130	www.giacc.com

Timeouts

ICMP: 15 seconds
TCP wait for ACK: 30 seconds
TCP: 600 seconds
TCP keep alive enabled: yes
UDP: 600 seconds
Wait after close: 20 seconds

Assignment 2f- VPN Security Policy Discussion

A Virtual Private Network (VPN) is a system that allows two private, secure networks to safely communicate via an insecure medium. In GE's case, the two networks are 1) the internal network of GE's headquarters and 2) the virtual network of a mobile employee's laptop and the insecure medium is the Internet. The design intent is to require strong authentication followed by strong encryption. Since the expectation is that both networks are secure, personal firewall and virus scanning software will be required on all laptops and such will be regularly audited to assure ongoing compliance. The implementation of VPN is based on the Internet Engineer Task Force (IETF) Internet Protocol Security (IPSec) standard. The VPN client is Windows-compatible and based on SafeNet VPN SoftRemoteLT, a widely used VPN client.

Authentication is a guarantee that the data received is the same as the data sent and that the sender represented is the actual sender. In the Mobile client VPN implementation used here, authentication is performed doubly.

1) Initially, by a proprietary program called GBAuth that requires correct user identity and password entry each time VPN communications is initiated by the mobile client.

2) Subsequently, by a pre-shared key authentication method (phase I) of the security policy

Confidentiality means that the receiver knows what was sent but unintended parties do not. Encryption is a method used to provide this confidentiality. IPSec provides this through a protocol called Encapsulating Security Payload (ESP) and this is implemented, along with key exchange, within phase II of the policy. The confidentiality provision of ESP can operate in two modes: Tunnel mode and Transport mode. Tunnel mode encapsulates the entire IP datagram within the ESP header including IP addresses and ports. Transport mode encapsulates just the transport layer frame inside ESP. The SafeNet VPN client SoftRemoteLT

implementation uses only ESP in tunnel mode. The GTA firewall performs the role of a security gateway by encrypting and routing packets. Access is allowed from a mobile client connecting at any registered Internet IP address. A unique IP address is assigned on the chosen virtual address segment on each employee's laptop. This virtual segment is configured as part of the GTA Firewall's mobile VPN policy. These implementation choices allow secure, full Network Address Translation (NAT) operability with diverse application support.

In this particular VPN implementation the Authentication Header (AH) protocol is not used. However, it should be noted that data origin authentication and anti-replay capabilities can, and in this particular VPN implementation are, accommodated within ESP's realm.

Before IPSec can secure an IP packet, a Security Association (SA) must be established through a special IKE mode that has been tailored for the GTA firewall's mobile VPN client. Identified by a unique IP Address, Security Parameter Index (SPI) and protocol (ESP, etc.), the SA specifies the parameters for communication.

Discussion of the actual VPN policy shown in Assignment 2e

Four passages in the GTA firewall configuration appear in red and pertain to the VPN policy that has been implemented for GE employee remote access capabilities.

The first passage is found in the **Users** section. One user, Sample, has been configured to demonstrate proof of concept. In order to authenticate using the GBAuth program, both an "identity" (suggested is to use user's email address to ensure a unique value—but something a bit more creative might be valuable) and a non-displaying password are specified. The "remote network" is actually a unique IP address on the virtual subnet 10.1.2.0/24. In Sample's case, the last octet of the IP address is chosen to be 1. The next user defined could be assigned 2, and so on.

The second passage is VPN Objects under the **Objects** section. The type is "MOBILE". GBAuth authentication is required. VPN packets are expected to arrive on the external interface. Force Mobile Protocol uses a dynamic IP addressing scheme in negotiation—we select "no" as we're using a static assignment of one address per user. The local network specified is those segments and associated systems we wish the VPN users to be able to access. Here notice that one CIDR specification includes the access of both the internal and service networks in one definition. Finally, the security policy choices are specified for the Authentication (phase 1) and Key Exchange (phase 2). A phase 1 exchange establishes a security association by negotiating the terms of the VPN, authenticating the validity of the VPN peer and setting the VPN connection parameters. This phase creates the initial security association database entry

used to conduct phase 2 exchanges. Phase 2 establishes security associations for the other protocols. Many of the following settings are fixed and therefore not seen in the VPN (red) sections of the policy shown in Assignment 2e.

Authentication/Phase 1
Negotiation mode=aggressive
Aggressive requires fewer exchanges of information / still fairly secure
Enable Replay Detection=yes
Authenication Method=preshared key
Encryption algorithm=3des
Hash algorithm=hmac-sha1
Perfect Forward Security Key Group=Diffie-Hellman Group 2
PFS determines how a new key is generated / prior used key cannot be used to derive additional key limiting consequence of one compromise
D-H Group 2 is the particular PFS key generation algorithm selected

Key Enchange/Phase 2
Security association life=200 seconds
Encapsulation protocol=ESP
Compression=none
Encryption algorithm=3des
Hash algorithm=hmac-md5 group 2
Encapsulation=tunnel
Authentication Protocol=no
Not using AH protocol / AH does provide stronger authentication but this typically inconsistent with NAT due to intolerance of IP header changes

Everything above must match in the remote client's policy configuration.

Remote access filters #4 and #5 represent the third VPN passage. Filter #4 allows inbound UDP port 500 (ISAKMP) for Key Exchange. Filter #5 allows inbound ESP protocol.

The fourth and final passage is IP Pass Through filter #2. Once packets have been de-encrypted at the core of the GTA Firewall, it is still necessary to explicitly allow them access to the internal and service networks without any further network address translation (NAT).

ASSIGNMENT 3- Verify the Firewall Policy

This assignment asks that the Firewall Policy be verified. This verification is sub-divided into planning (Assignment 3a), conducting (Assignment 3b), and evaluation sections (Assignment 3c). Assignment 3d offers recommendations for improvements and alternatives as requested in Assignment 3. I also decided to add two additional sections: 3e-Likely "requests" from employees, partners & suppliers and 3f- Other threats, other measures in an effort to more completely cover the perimeter security topic.

Assignment 3a- Planning the Validation

The validation should be meticulously planned defining needed equipment and programs, initial and final conditions of all equipment and connections, sequencing of tests, etc.

The actual evaluation should be conducted in GE's weekly global maintenance window of 17:00 to 23:00 GMT Sunday. This particular period presents a minimal business consequence even if there's a serious degradation or outage.

It is envisioned that the staffing for the evaluation will be comprised of salaried employees. Personnel costs will just amount to the unavailability of these individuals during those periods of time devoted to the three phases of this evaluation project. The tools used will be entirely open source ones that will incur no additional costs. I would recommend that two people be assigned to the task. That way, the work has the benefit of being crosschecked. The time estimates in man-hours are as follows: planning=6, conducting=12 and evaluation/report presentation=32. If outsourcing were considered, rates of $200 US dollars per hour would not be unexpected. This would amount to (6 + 12 + 32) x $200 or about $10,000 US dollars for the verification.

The worst thing that should be expected of the verification process would be that:

1) The firewall becomes inoperative; passing all traffic.
2) Internal and/or service network systems could become compromised such that significant rebuilding & restoring is required.

It is suggested that all but the simplest systems remain shutdown or disconnected from the network during the earlier or lighter-duty verification stages. Then, as confidence grows, these systems can gradually be brought online. The Sunday maintenance period has already been selected to mitigate the risk of business interruption should something unforeseen occur.

Assignment 3b- Conducting the Validation

The validation requires at least two tester laptop systems. Each of these systems should be equipped with the following programs: ping, traceroute, telnet, tcpdump and nmap. All of these programs were included in my RedHat Linux 7.3 distribution. In addition, I referenced systems as destinations (one on each segment) that included TELNET and TFTP servers. This exposes TCP port 23 and UDP port 69 for test purposes. The system on the service network representing the public web server (www.giacc.com) also has tcp ports http and https open.

I will start using single upper case letters in my discussion to represent points on the testing sketch shown on the next page. B, C and E represent the GTA Firewall's interfaces. A, D and F represent other systems on each of these segments. G represents the mobile VPN client. Two letter (from-to) pairs will indicate tests. For example, AB would signify a point A system directing test packets to destination B.

The first validations should check traffic (ICMP, TCP and UDP) targeted at each of the firewall's three interfaces from a tester laptop. This will be accomplished by running ping, traceroute, and nmap. Ping validates ICMPs, traceroute (run with a TTL value of one validates the TTL=0 response of the firewall itself, and nmap validates TCP (SYN scan) and UDP characteristics. No protocols or ports should be open unless expected and necessary. AB, DC, and FE tests should be performed for each of these tools.

A second stage of the validation should check the rulebase to assure it is operating as configured. Here, a tester laptop is attached as A, D, and F to perform ping, telnet, and nmap (TCP SYN and UDP scans again) tests AD, AF, DA, DF, FD, FA. Telnet checks the allowance of a TCP connection. A second tester laptop, running tcpdump, is to be attached to the target/destination segment in a promiscuous sniffing mode. Only packets consistent with the security policy and rulebase should be observed.

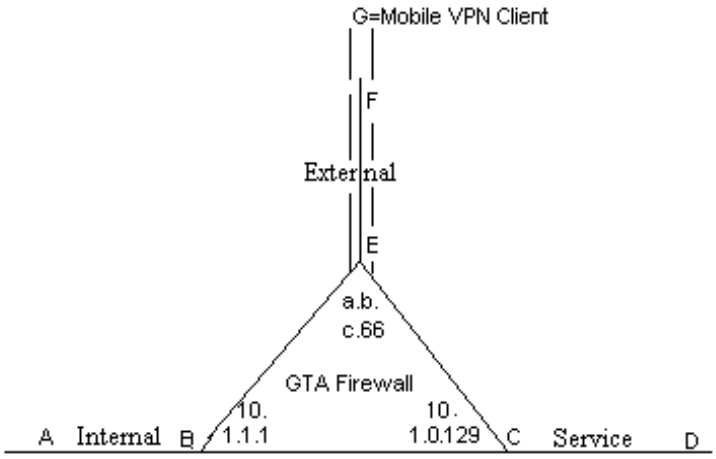
Other tests should validate the access of G, an authorized and successfully initiated remote VPN client. Here, the VPN client system itself performs ping and telnet tests GA, GD and GE. I've also included tests from an external source (F) to the destination of the public web address, www.giacc.com, as referenced by the registered IP address of a.b.c.130 and denoted in the testing labels as D'. Also shown in the results table are pings from the firewall itself.

These additional items should also be verified:

- Our policy requires that mobile VPN users authenticate, prior the being allowed VPN access. If VPN operations should be commenced without this prior GBAuth authentication, will it be successful?

- When VPN operations have been successfully engaged, are the supposedly secure packets really encrypted? Verify with tcpdump.

- Finally, It is important to verify that the preceding tests have generated sufficient and meaningful firewall log records.



Assignment 3c- Evaluating the Results

Results Table

```
ABP x/r4          DCP x/r4          FEP x/r4          GEP x
  I x/r4            I x/r4            I x/r4            I x
  T x/r4            T x/r4            T x/r4            T x
  S r5              S r9              S r10
  U r6              U r9              U r9

ADP o             DAP x/r2          FAP x/r1          GAP o
  T o               T x/r2            T x/r1            T o
  S r7              S r9              S r9
  U r8              U r9              U r9

AFP x/r2          DFP x/r2          FDP x/r1          GDP o
  T x/r2            T x/r2            T x/r1            T o
  S r9              S r9              S r9
  U r9              U r9              U r9

AGP x/r3          DGP x/r3          FGP x             GFP x
  T x/r3            T x/r3            T x               T x

BAP o             CDP o             EFP o             EGP x

r1=illegal forward attempt logged         FD'P x/r4
r2=outbound filter block logged           FD'T x/r4
r3=passthrough filter block logged        FD'S r11
r4=remote access filter block logged      FD'U r9
```

```
r5
# nmap (V. 2.54BETA31) scan initiated Mon Oct 27 11:22:40 2003 as: nmap
-v -g53 -sS -PT -P0 –T 3 -o abs.txt p1-65535 10.1.1.1
Interesting ports on  (10.1.1.1):
(The 1551 ports scanned but not shown below are in state: filtered)
Port       State       Service (RPC)
77/tcp     open        priv-rje
443/tcp    open        https
2784/tcp   open        www-dev
# Nmap run completed at Mon Oct 27 11:27:55 2003 -- 1 IP address (1
host up) scanned in 315 seconds

r6
# nmap (V. 2.54BETA31) scan initiated Sat Nov  8 06:42:24 2003 as: nmap
-v -g53 -sU -PT -P0 -T 3 -o abu.txt p1-65535 10.1.1.1
All 1459 scanned ports on  (10.1.1.1) are: filtered
# Nmap run completed at Sat Nov  8 07:12:40 2003 -- 1 IP address (1
host up) scanned in 1816 seconds

r7
# nmap (V. 2.54BETA31) scan initiated Mon Oct 27 12:38:05 2003 as: nmap
-v -g53 -sS -PT -P0 -T 3 -o ads.txt p1-65535 10.1.0.130
Interesting ports on  (10.1.0.130):
(The 1551 ports scanned but not shown below are in state: closed)
Port       State       Service (RPC)
23/tcp     open        telnet
80/tcp     open        http
443/tcp    open        https
# Nmap run completed at Mon Oct 27 12:38:57 2003 -- 1 IP address (1
host up) scanned in 52 seconds

r8
# nmap (V. 2.54BETA31) scan initiated Mon Oct 27 17:13:05 2003 as: nmap
-v -g53 -sU –PT -P0 –T 3 -o adu.txt p1-65535 10.1.0.130
Interesting ports on  (10.1.0.130):
(The 1458 ports scanned but not shown below are in state: closed)
Port       State       Service
69/udp     open        tftp
# Nmap run completed at Mon Oct 27 17:14:03 2003 -- 1 IP address (1
host up) scanned in 58 seconds

r9
# nmap (V. 2.54BETA31) scan initiated Mon Nov 10 11:58:53 2003 as: nmap
-v -g53 –sS(or U) -PT -P0 -T 3 -o /various.txt p1-65535 various
All 1554 scanned ports on  (various targets) are: filtered
# Nmap run completed at Mon Nov 10 12:27:22 2003 -- 1 IP address (1
host up) scanned in 1708 seconds

r10
# nmap (V. 2.54BETA31) scan initiated Sat Nov  8 07:18:12 2003 as: nmap
-v -g53 -sS -PT -P0 -T 3 -o /fes.txt p1-65535 a.b.c.66
Interesting ports on  (a.b.c.66):
(The 1552 ports scanned but not shown below are in state: filtered)
Port       State       Service
25/tcp     open        smtp
76/tcp     open        deos
```

```
# Nmap run completed at Sat Nov  8 07:26:50 2003 -- 1 IP address (1
host up) scanned in 518 seconds

r11
# nmap (V. 2.54BETA31) scan initiated Thu Nov 13 14:23:48 2003 as: nmap
-v -g53 -sS -PT -P0 -T 3 -o fd-s.txt p1-65535 a.b.c.130
Interesting ports on a.b.c.130:
(The 1552 ports scanned but not shown below are in state: filtered)
Port       State       Service
80/tcp     open        http
443/tcp    open        https
# Nmap run completed at Thu Nov 13 14:34:54 2003 -- 1 IP address (1
host up) scanned in 666 seconds
```

Results Table Description

The results table is intended to display the test results in a compact and easily understandable form. The first two characters of each entry represent the from and to systems corresponding to the labeled points on the testing sketch. The third character represents the test type: P for ping, I for ICMP, T for telnet, S for Nmap SYN scan, and U for Nmap UDP scan. Following each test code is the result. “o” means communication/access succeeded. “x” means that there was no response or connection achieved. Finally “r” followed by a number indicates a more detailed result. In some cases, these show firewall log entries; in others, Nmap output. Result 9 is a particularly common one of all ports being filtered. The exact nmap command line arguments are shown in the results.

Nmap (http://www.insecure.org) is a tool used to scan hosts in an effort to determine what ports are “open” or in receiving mode. The output shown in the results above is self-explanatory. However, I would like to add that it is possible (although there is no instance seen here) to have a port listed as “closed”. This would be the case, for example, in a TCP SYN scan where the system is reached with the SYN packet but the port is not “listening” and a RST is returned. A SYN|ACK return would be indicative an open port. No response would be declared filtered.

Here’s a brief description of the command line options used:

-v	Verbose mode	More information is presented in the output
-g53	Source Port 53	Many firewalls give port 53 (DNS typically) a special allowance to establish a connection even if no rule
-sS	TCP SYN scan	A TCP SYN packet is sent to each port & the reply is noted: SYN|ACK means open, RST means closed
-sU	UDP scan	A zero byte UDP packet is sent to each port; if reply is ICMP port unreachable, its closed; otherwise its open

-PT	TCP ping	Use TCP ACK packets instead of ICMP echo packets to determine what hosts are available for testing.
-P0	No ping	Don't bother to ping hosts before scanning them
-T 3	Timing policy	Normal; runs as quickly as possible without overdriving
-o	Write output to <filename>	
p1-65535	ports	test with ports 1 through 65,535

When conducting the AD, AF, DA, DF, FD and FA tests, no unexpected packets were observed on the destination segments using tcpdump. Mobile client VPN operations commenced without first authenticating did fail with a log entry of "denied—GBAuth authentication required". Tcpdump did capture UDP ISAKMP and ESP (protocol 50) packets associated with the VPN usage. Finally, log entries were abundant and meaningful. All in all, what was expected was observed and what shouldn't have been observed wasn't. An example of the latter was that UDP packets that weren't supposed to be passed from one interface of the firewall to the other were logged as blocked and Tcpdump confirmed were indeed blocked. Also, the TTL decrementing to zero by the firewall produced no ICMP messages.

I feel compelled to say that much more was observed than was being looked for at any moment. To mention just a few, I saw DNS, Network time calls, and log entries being ferried to the syslog server.

A SANS/FBI Top 20 Vulnerabilities Scan by Qualys, Inc run from the Internet found no vulnerabilities.

Assignment 3d- Recommendations for Improvements and Alternatives

The security policy designed and implemented here specifically was kept as simple and restrictive as possible. This section discusses items that should be considered to provide additional robustness and resiliency. In the name of redundancy, having a backup or spare for everything would be a good idea. A spare border router, a spare firewall, a second T1 line (perhaps use two (vendor diverse)) and develop a way to load share the lines whenever both are operational. All the servers should be able to be rebuilt perhaps on a generic spare platform within a certain (short) amount of time. I allowed for only a single ISP mail relay server. Best practices would suggest providing for an alternate or two of these relays.

I was wondering whether the benefit of the extra GBAuth authentication for mobile VPN clients exceeds the exposure of having TCP port 76 exposed to external sources. It might be a good investment to issue all employees authentication token devices. These are small enough to attach to a key chain, prompt the user for a PIN and respond with time-based code sequences. These codes, which are synchronized with an internal radius type authentiction server, are required for VPN authentication.

The GTA Firewall came with a default rulebase such that most of the "allow" rules were deleted because I didn't see them referenced. One rule allowed TCP port 113 has to do with authentication and identification. Apparently, some email servers require a response on this service before passing email. I didn't notice a case of this in practice, but best practices may beg such a rule be returned to active policy. Experience would tell if there are other such instances.

Both performance and single-point-of-failure reasons could make a case there perhaps should really be a separate http proxy server. Similarly, perhaps an inbound http or reverse proxy system should be used to further protect the web server. In both these instances, simplicity is balanced with spreading exposure and load across additional systems.

Although administrative access of the GTA Firewall is encrypted and restricted to only internal network systems, further restricting access to just the console interface (as was done with the border router) should be considered to further improve security.

3e- Likely "Requests" from Employees, Partners & Suppliers

Employees may request the capability to internally use FTP clients to Internet-based FTP servers. Partners and Suppliers may ask that GE host an FTP server (ftp.giacc.com). Proper business justifications would need to be made for both cases. In each, I would add a system to the service network. In the first case, this server would run an FTP proxy service such that the client would reference the FTP Proxy service server and provide the actual server name or IP address as well as and an ID and password. The proxy would then be allowed Internet DNS and FTP service port references to be able to satisfy the client requests. Internal client access would also be added to the firewall's policy

A hardened, separate Linux system would run the (ftp.giacc.com) FTP server. FTP access from the both internal and external network would be supported, modeling the public web server's firewall configuration.

Special Ports and Provisions
It is likely that requests will follow the original design for special ports and provisions. Examples of these might be for AS/2 (EDI transactions), Internet-based Citrix servers, multi-media for online training, etc. Again, business case justifications should be required. If the implementation necessitates an external initiation (partner or supplier clients coming to a GE server), the GE server should be placed on the service network. A remote access rule as well as the triad of NAT settings (Alias, Inbound Tunnel, and Static Address Mapping) would need to be configured. Model these settings on the current www.giacc.com (web server) policy. In those instances where GE employees initiate the access (operate the client), the requests should be able to be handled simply through the addition of an outbound filter.

Keep SSL/Encryption in mind
If possible, try to incorporate SSL (Secure Sockets Layer) aspects into the solutions of access requests. Both open source and Windows platforms have offerings such as Stunnel (**www.stunnel.org**) that allow the encryption of arbitrary TCP connections within a SSL tunnel. This allows you to secure non-SSL protocols such as IMAP, LDAP, etc. without needing to alter the base program or protocol.

3f- Other Threats, Other Measures

Spyware

Spyware, software loaded onto a user's system with or without their consent that passes information back, represents a growing threat. Spyware may be purchased or downloaded (for a given desired functionality). It may come attached to various utilities, media players, games, and other shareware and freeware applications. Spyware can also be obtained by simply accessing a web site and clicking on certain yes boxes the popup. Because users often fail to read the "fine print" and disclaimers, the information-gathering aspects often go unnoticed. Spyware can be classified into two categories: advertising (sometimes called adware) and surveillance. Advertising spyware typically provides users with targeted pop-up ads based previous web pages referenced. Information captured and returned includes the user's first name, country, city, zip code, online buying habits, what software is on the computer, certain web pages viewed, etc. In addition, adware just plain compromises the system and the network to which it is attached. By allowing the capability of things like automatic software updates, the systems affected are more susceptible to the infiltration of malicious items from any source. Network resources are being consumed by totality of the inbound unwanted and the outbound captured information and the number of clients affected multiplies this!

Surveillance spyware, increasingly sophisticated and stealthy, goes beyond intrusion, manipulation and commerce. Its purpose is to steal information or monitor access depending on intent. One product in this category, keyloggers, record what's typed, certain screen shots, and even passwords. Such programs are sold commercially to monitor and control employees (by companies) and children (by parents). RATs (either Remote Administration Tools or Remote Access Trojans depending whether considered good or evil) not only provide remote monitoring and recording but control as well. Again, a range of commercially available administration tools as well as malware Trojans and worms such as Sobig (intentions definitely bad) all function with the same principles. For a variety of reasons anti-virus products frequently turn a blind eye and a deaf ear toward spyware. Various programs and tools are available to detect and combat spyware such as Spybot (http://www.safer-networking.org) and Ad-aware (http://www.lavasoftusa.com).

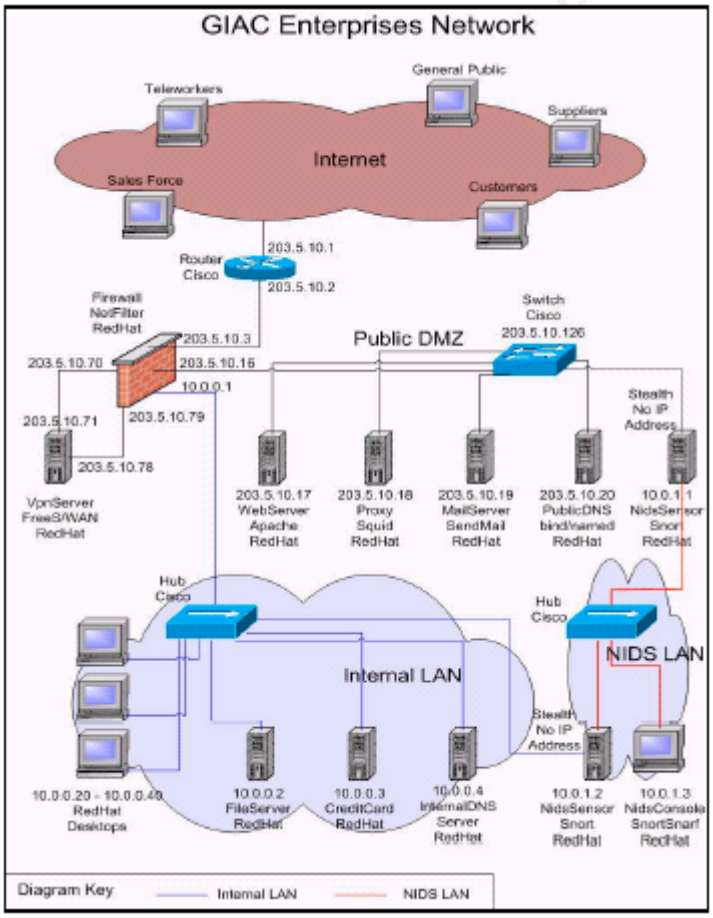
Application-level attacks

Increasingly, attacks on networks are being embedded in the application traffic flow allowed to pass between clients and servers. Stateful inspection firewalls, mostly operating at the network level, are not designed to look into or interpret application-level data. Application-specific firewalls, employing proxy technology are typically tuned to one application. Here the proxies understand application logic and can accept or deny packets based on this knowledge. Other solutions new to the market are Intrusion Prevention Systems (IPSes). These devices examine the messages of several applications and understand enough of the protocol (through pattern checks on service field values, behaviors, statistics and attack strategies) to be able to make legitimacy decisions that go beyond network-level analysis. IPSes are controlled by rulebases to specify the exact responses desired for various attack event triggers and can be updated with specific signatures of prominent attacks.

Whatever systems are deployed to guard against these new application-level threats, the difficulty of the task shouldn't be underestimated. There's a great deal of knowledge of application operation and the mechanisms involved. How is just the pertinent application-level information extracted from packets for inspection? How exactly is this information analyzed to determine if it is fair or foul? Every vulnerable application needs to be protected and this is expected to play an increasing role in the future of network security.

ASSIGNMENT 4– Design Under Fire

For assignment four, I have selected the practical prepared by David Jenkins in July 2003. http://www.giac.org/practical/GCFW/David_Jenkins_GCFW.pdf



The border router selected here is a Cisco 1760 running IOS 12.2 and the firewall is Netfilter/IPTables 1.2.8 implemented on RedHat Linux 8.0. Linux seems to be the exclusive OS involved here for all corporate servers and workstations/laptops. The graphic above depicts David's main systems and connections. He stated in his practical that "Netfilter is not a simple firewall to use…and it is recommended for someone with the necessary technical skills…" Such an admission seems to imply that there might be unrealized vulnerabilities and exposures. Was the Linux OS really hardened completely and sufficiently? Have any unnecessary and exposing processes or ports been overlooked?

Three attacks will be proposed to defeat this design: a direct assault on the firewall itself, a distributed denial of service attack (this affecting predominantly the border router) and a plan to compromise an internal system.

Assignment 4a– Assault the firewall

The target is the firewall itself.

Firewall vulnerability

A Linux Kernel 2.4 vulnerability has been identified which can be exploited to cause a Denial of Service condition. (http://www.secunia.com/advisories/8786/) The defect lies in the manner the Linux Kernel handles caching of routing information. The handling of this cache will consume large amounts of CPU resources if the system is presented with a stream of spoofed source address packets. Such could put an iptables (netfilter) system in an out of service condition with a rate as low as 400 packets per second. The actual rate is dependent on the exact spoofed source addresses selected. Those that cause the greatest number of hash collisions in the cache table produce the maximum impact at any given packet rate.

The syntax of many of the extended access list statements shown in David's border router policy is not valid. So it isn't clear to me what, if any, screening of spoofed source addresses would actually be performed. I'm speaking of the statements of the form "access-list 101 deny ip-addr/mask-bits le 32".

Linux networking programs use various hash tables to classify packets into caches. Two notable tables should be of concern: the Netfilter connection tracking table and the route cache. The latter, used to make routing decisions involving traffic flows, is of particular interest here. Cache matches are organized based on source address, destination address, and TOS value. The Type of Service (TOS) field in the TCP header is a one-byte value used to specify the quality of service, datagram precedence, reliability, etc.

The actual vulnerability lies in the cache hashing algorithms of traffic flow entries. Carefully chosen source addresses and TOS values can be used to produce hash function collisions such that distinct flow entries are required to be generated for each crafted packet received. The effects of the attack are increased as the routing cache size is increased. By default, the more physical RAM installed on a platform, the larger the routing cache memory allocated. However, this value can be explicitly set.

If sufficiently adept at solving a small system of linear equations over the field of two elements, an attacker can determine and send packets to take particular advantage of this vulnerability. Unfortunately, my research found no available Internet source of the particular exploit tools recommended or the commands used in such an attack. Runaway CPU usage could have many causes, but a troubleshooter might soon try disconnecting the interface connecting to the Internet through the border router. This would suspend the symptoms and confirm that the source of the problem was somehow packet traffic from this interface. However, since spoofed source addresses are used, I would suspect discovering the specific attacking source system would present a challenge.

Countermeasures

A patch has been produced that changes the hash function from linear to non-linear. Such a change adds sufficient complications to thwart this just discussed exploit. This same RedHat issued patch also addresses a similar vulnerability in the Netfilter connection-tracking table. If, for some reason, the patch can't be applied, there are two adjustments that can mitigate the effects of the exploit.

1) Setting Netfilter rate limits

2) Reducing the routing cache size using the /proc interface
   #echo 4096> /proc/sys/net/ipv4/route/max_size
   #echo 2048> /proc/sys/net/ipv4/route/gc_thresh
   (edit /etc/sysctl.conf to make changes permanent)

Use caution in reducing the cache size as this can have a severe impact on performance if the number of concurrent flows exceeds this maximum.

Assignment 4b– A Distributed Denial of Service Attack

TFN2K is designed to simultaneously launch denial-of-service attacks from several sources against a set of targets. It is designed to be difficult to recognize and filter and to camouflage the source by spoofing IP addresses. TFN2K can forge packets that appear to come from neighboring systems. All transmissions are unidirectional (for example, attack commands are sent from the server to the clients 20 times to assure delivery since acknowledgements and responses have been rendered inoperative by source address spoofing). Multiple transport protocols are used including UDP, TCP, and ICMP. TFN2K's strategies include sending large amounts of data to overwhelm its victims and the capability to crash or introduce instabilities by sending malformed or invalid packets. TFN2K uses a client-server architecture in which a single client, under the control of an attacker, issues commands simultaneously to a set of TFN2K servers (unprotected, compromised, unaware accomplices). The servers then conduct the actual denial-of-service attacks against the victim(s). With the objective to be as stealthy as possible, TFN2K client to server communications can use spoofed source addresses, be sent via randomized TCP, UDP and ICMP packets, be encrypted and mix in non-essential decoy packets. Here, an enormous leverage is gained by having so many systems directing packets at one or a few targets. One source of TFN2K can be found at ftp://ftp.ntua.gr/pub/security/technotronic/denial/

The first step in actually carrying out this scheme is to identify 50 Cable/DSL connected systems that can be compromised and used as "attack" servers. If I traceroute from my own cable service, I can see that the next upstream router is named c-x-y-z-1.wash. client2.cableco.com at IP address x.y.z.1 (x, y and z substitute for the actual ip address octet integers. I know my IP subnet mask is 22 bits, so I try to see if there's an x.y.z+4.1, an x.y.z+8.1, etc. Confirmed. Now, I methodically use a tool like www-arc (www.nessus.org) or SARA (www-arc.com/sara) checking each candidate IP address to identify vulnerable systems I can compromise. Is it expected that many, if not most systems will not use a DSL/Cable router and will not have a personal (software or hardware) firewall. Nessus or SARA will report specific vulnerabilities for each system. www.google.com can then be searched for exploits to match each vulnerability. As exploits are downloaded targeting to each new specimen, it would be prudent to launch Nessus, in turn, from each newly compromised system. This helps to mask the ultimate source of these nefarious activities.

After 50 (or more—just in case some are not available or do not participate when needed) systems have been made receptive, TFN2K server code (attack agents) will be uploaded and run. Tools like those offered by www.smart-hack.com/sub7.html can be used for this purpose. At this point these "servers" are primed to accept attack commands. Our favorite of the compromised systems will be selected for special duties as the TFN2K server.

Target selection

It is now time to list the IP addresses and ports of good targets (packets destined such that firewall policy insists on passage and proper delivery):

1)	Web Server	203.5.10.17	tcp	80
2)	Mail Server	203.5.10.19	tcp	25
3)	Public DNS	203.5.10.20	udp	53
4)	VPN Server	203.5.10.71	udp	500

The commands to use are “commence UDP flood” and “commence SYN flood, port %s”. In addition to straight TCP SYN and UDP floods, TFN2K offers an ICMP echo flood, an ICMP broadcast (SMURF), a “mix” flood, and even a bogus/malformed/invalid packet attack variation to choose from. The command for the latter is “commence targa3 attack”.

Any manipulation of any of these systems could potentially be noticed. However, there is enough spoofing of source addresses, compromising subsequent systems from prior systems compromised, decoy packets, only unidirectional delivery, etc. to make detection of the instigator rather unlikely. Lack of participation of a few of the compromised systems should be anticipated and inconsequential.

Countermeasures to mitigate attack

TFN2K uses TCP and UDP (both to random ports) as well as ICMP to DoS its victim. Although difficult to stop, here are some measures that would render such attacks less effective:

1) Block unnecessary ICMPs and especially externally initiated ones in the border router access-list. Typically only ICMP type 3 (destination unreachable) packets should be allowed.

2) Consider Cisco IOS rate limit statements.

3) Use commercial products such as Top Layer’s Attack Mitigator™ IPS. This is a high-speed, inline product capable of detecting and blocking such attacks.

Assignment 4c– Compromising an Internal System through Perimeter

The decision of which internal system to target among those presented in David's design is not an easy one to make. However, the Credit Card server seems like a good bet since the information is of direct (illegal) value. Such a compromise could cause great damage to the company. A disadvantage of this choice is that Swatch is running on the server and our activities might be noticed more readily and sooner than on other systems.

The compromising of an internal system from outside the perimeter is particularly challenging. The effectiveness of external reconnaissance and tools such as nmap (www.insecure.org/nmap), SARA (www-arc.com/SARA), and Nessus (www.nessus.org) are hampered by the typical firewall behavior of network address translation (hideNAT) and state table blocks of most externally originating packets that aren't expected response packets. Tools like p0f (http://www.stearns.org/p0f) could still be used make certain conclusions about internal systems as long as these systems weren't using some kind of intermediate proxy. In that case, we'd have profiles and fingerprints of the proxy system (or firewall) rather than the actual target. We are blessed with the internal architecture details from David's write-up (which is probably beyond what could be determined from perimeter reconnaissance). However, a scheme to deposit something unwanted or alter things to suit our interests directly through the perimeter to an internal system of another certification candidate's design would seem to particularly improbable. With all we've learned, indirect attack proposals seem to me to be the most realistic and have the greatest chances for success.

My first "indirect" idea was to try to first compromise a DMZ server (such as the WEB server) and then to direct this system to subsequently mount an attack against the target. Surely, there must be some transactions (very tightly constrained) from the DMZ Web server to the Credit Card server already allowed. However, with two hardened Linux servers and two trips through firewall involved, the prospects for success are outweighed by the likelihood of detection.

My second--and I believe better "indirect" idea was to penetrate the perimeter into the internal realm though radio waves. Wireless networking is increasingly common to find in today's companies. Any wireless access point attached to an internal network segment essentially bridges that internal network to surrounding radio waves! Consider the penetration and compromise opportunity with both the border router and firewall taken completely out of the path! Long-range antennas that can receive signals from up to 2000 feet away are available commercially or can even be home built (http://www.oreillynet.com/cs/weblog/view/wlg/448). Intrusions can be launched while you and your equipment are completely out of sight and relatively safe. I propose the use of a tool called kismet (http://kismetwireless.net).

Kismet is an 802.11b wireless network sniffer that separates and identifies all components operating in any given venue. It is open source, publicly available, particularly adapted to Linux and operates with most rfmon support capable wireless adapters. It is best to run kismet as root and you'll need to tailor the kismet.conf and kismet_ui.conf files to match your adapter and setup. These files use an option=value format and all of the options are documented by the text found within these configuration files. Kismet.conf controls the sniffer operation such as capture sources, logging options, WEP decrypting, etc. An example of the syntax for capture sources (usually there's only one wireless adapter) is "source=cisco,eth0,Cisco" signifying a Cisco card on interface eth0 named "Cisco". Certain packets logged to the file can be eliminated by applying filters: "noiselog" (too short or otherwise unfit packets) and "beaconlog" (multiple beacons with the same network and SSID) and "phylog" (physical layer packets like data acks, etc). If WEP is in use and you have cracked the encryption key (programs for this purpose: WEPCrack and AirSnort are briefly described following the discussion of kismet), the wepkey=keyvalues option will instruct kismet to perform on-the-fly decryption. The kismet_ui.conf controls user interface aspects such as columns (clientcolumn option) and colors (xxxcolor option). Kismet displays three primary panels or views: Network display (shows discovered networks), Information (shows packet counts, packet rate, elapsed capture time, etc), and Status (shows events and alerts). There are other "popup" displays that can be bought into view as well and all of the presentation options are dynamically controlled through a list of single character commands.

AirSnort (http://sourceforge.net/projects/airsnort) and WEPCrack (http://wepcrack.sourceforge.net) are open source programs that take advantage of Wireless Encryption Protocol (WEP) weaknesses. These tools passively monitor transmissions until gathering sufficient packets to be able to recover the encryption key. Typically one key is used for the entire LAN so this would allow an intruder significant participation is a supposedly WEP-secured network.

Attack proposal

The first step in an attack would be to passively learn all we can about the internal network through the use of kismet as described previously. Step two is to use this information to become a wireless client on the network and run nmap to scan the Credit Card server. The information discovered in steps one and two will determine the precise strategy to follow in subsequent steps. However, the plan going forward would generally involve a combination of disabling legitimate access points or disabling active client associations with legitimate access points. At the same time, we would reconfigure some of our equipment to masquerade as a legitimate access point luring unsuspecting stations to connect! Once a client accepts our access point, there are proven tools available to exploit the system to suit our interests. For example, the code and instructions for a Credit Card server exploit could be installed and scheduled when we are no longer on (or nearby) the site. We could then check with this client on another occasion, collect illicit data, or give further instructions.

One constraint on our attack is that it is common for companies to attempt to "secure" their wireless LAN with authentication based on a list of valid MAC addresses. Kismet will report these addresses but it may be necessary to change our own equipment's MAC address to one operating legitimately in order to authenticate. Unless we somehow disable the original system, duplicate MAC addresses might be detected and network personnel alerted. The latter would depend on what, if any, wireless intrusion detection systems are in use.

Unless they physically spot our surveillance vehicle and investigate, the kismet phase should be completely undetectable as we are only listening to radio waves. Once we "join" the network as a client or masquerading access point, detection will be a distinct possibility.

Ironically, the best countermeasure for this wireless intrusion proposal would involve the same tool we've been discussing. Kismet's value is not restricted to eavesdropping and detail reporting of someone else's network. It can serve as an excellent wireless IDS (Intrusion Detection System). Kismet can generate various alerts when it finds traffic that match certain defined attack characteristics or other anomalies. It can be made to pipe data into more traditional IDS systems such as Snort. Also, remote drone probes that "listen" in remote wireless venues can be interconnected to achieve a more complete monitoring system.

If wireless is not in use in the office, employee homes could be checked. Follow them home with a VAN capable of Kismet surveillance! Wireless is increasingly popular on the domestic front and implementations tend to be even less secure! We're looking for an opportunity where a VPN client equipped company issued laptop is left operational on the home wireless network without the VPN tunnel to the office actually enabled. Also, a disabled or non-existent personal firewall on this system would be convenient. If conditions were favorable, the laptop could be passed a Trojan program that would become active once the VPN to the office was enabled or when the laptop is docked back at headquarters.

The last resort "indirect" plan, riskiest and most extreme, would be to physically attach an entire Trojan system inside on an internal network segment. Security guards, the cleaning crew, or contractors could be motivated to attach a well-concealed, small system to the internal network. This would have to be made of untraceable components and use a great deal of encryption and stealth in its operations. It would have to be constructed to reveal very little about its master if discovered! Perhaps it could be programmed to self-destruct after a period of time or if disturbed. It would probably not be wise to have it attempt packet transmission to the Internet as such would be likely be logged. Internet destination information would be particularly damning. After a recognizance collection phase, perhaps the unit could be shutdown, removed from its dark corner and returned for analysis.

References used in addition to those cited throughout the text

Gilbert Held and Kent Hundley, Cisco Access Lists Field Guide. McGraw-Hill, ©2000.

ICSA Labs Modular Firewall Certification Criteria
http://www.icsalabs.com/html/communities/firewalls/certification/criteria/criteria_4.0.shtml

Global Technology Associates, Inc, GNATBox VPN Feature Guide, 1996-2002

Global Technology Associates, Inc, GNATBox User's Guide, 1996-2002
http://www.gta.com/support2/documents/

Cisco 2600 TCP SYN rate limiting
http://www.cisco.com/warp/public/707/newsflash.html

Scott A. Crosby, Dan S. Wallach, Denial of Service via Algorithmic Complexity Attacks
http://www.cs.rice.edu/~scrosby/hash/CrosbyWallach_UsenixSec2003/index.html

Mike Kershaw, KISMET 3.0.1 Documentation
http://kismetwireless.net/documentation.shtml

Clearswift Ltd., Beware Spyware ©2003
http://www.clearswift.com/news/whitepapers.asp

NetScreen Technologies, Inc. The Need for Pervasive Application-Level Attack Protection ©2003 http://i.n102.net/netscreen0001/h/dif_pdf.html