Global Information Assurance Certification Paper

Perimeter Protection using Sidewinder

GIAC Firewall Practical Assignment v1.6

Section 1

Security Architecture

GIAC Enterprises buys and sells fortune cookie sayings. The business relies heavily on its Internet presence to sell fortune cookie sayings from its web site. It replenishes it's stock by purchasing fortune cookie sayings from authors scattered all over the world. GIAC also sells to international business partners who translate and resell the sayings overseas. This section describes the Security Architecture that provides access control for the three types of business users that GIAC's network services.

- Customers buy fortune cookie sayings from GIAC Enterprises. They place orders on GIAC's public web server, which is shown on figure 1.
- Suppliers are authors that sell fortune cookie sayings to GIAC. Each has a unique client-to-gateway VPN tunnel defined in the Intel 3130VPN gateway
- Partners purchase fortune cookie sayings from the public web server, then translate and resell them overseas from their own domain.

Network Overview, and Traffic Flow

Refer to Figure 1

A Cisco router (model 3640) running the firewall feature set known as CBAC, and a Sidewinder firewall made by Secure Computing protect GIAC's network. The router does the initial screening and filtering of traffic. The firewall further inspects filters, and segments the traffic. The network has three segments. The GIAC internal subnet, which houses the mail server, the order database, the sayings web server and database, and a syslog monitoring station, along with a primary domain controller, (PDC) backup domain controller, (BDC), and several Windows 2000 workstations. There is a DMZ (demilitarized zone) for the public web server, and the partner's web servers. The third segment is for VPN traffic, which is used by fortune sayings suppliers. This VPN segment consists of the VPN gateway device, and a VPN burb on the firewall. The VPN burb is essentially a network card and subnet that is separate from the DMZ and GIAC internal subnets. A burb is Sidewinder terminology to describe a network segment. For more on burbs, see Section II, page 14.

Customers from the Internet are directed to the Internet burb on firewall. (see figure 1) From there, the firewall proxies them to the DMZ burb. Once on the DMZ burb, they can reach the public web server and place orders for fortune cookie sayings using a web server application. Partners also come in from the Internet, are directed to the DMZ, and place their orders for product on the partner web server, using the same web interface method as general customers. The two web servers query and update the customer and partner databases respectively using SSL. Secure Socket Layer is a tunneling protocol developed originally by Netscape. It uses digital certificates to verify and authenticate secure channel communication between the customers' web browser and the web server,

and between the web servers to the databases using the SSL protocol, also known as https.

Suppliers are independent contractors who work from home. As part of their service contract agreement with GIAC, they are directed to download the VPN client software and a PowerPoint tutorial on how to install it from GIAC'S public web server. Once the VPN client software is installed, they call GIAC to receive the pass-phrase and other parameters needed to establish an encrypted VPN tunnel. A GIAC system administrator verifies their identity before providing the requested. Suppliers can then run the client software to create an encrypted virtual private network tunnel across the Internet to GIAC's VPN gateway at the external (E1) interface. The traffic is decrypted then released at the internal, trusted side of the VPN gateway (E0). This unencrypted supplier traffic then hits the Sidewinder firewall VPN burb. The Sidewinder has an access control list (acl) that only permits web traffic on ports 80 (http) and 443 (https). Sidewinder's application level proxies pass this traffic to the internal, GIAC internal subnet burb. From there, the supplier uses his web browser to enter new sayings into the sayings database by way of a supplier web site. This is accomplished by a web server application that queries and updates the sayings database. The supplier's web server and sayings database are located on two different boxes on the GIAC internal subnet. All databases reside on the GIAC internal subnet. Each database query and update between the DMZ and VPN burbs to the GIAC internal burbs can pass only by Sidewinder's https proxy.

The servers are running NT 4.0 with Microsoft Internet Information Server 4.0 (IIS). All servers and workstations have been secured and hardened by the process described in the next paragraph.

NT Server and Windows 2000 Hardening

After the initial build, Service Pack 6a. is installed, then the NT 4.0 Post-Service Pack 6a Security Rollup Package installed to add in all the security updates released for NT 4.0 since the Service Pack 6a was built. A subscription to Microsoft's Product Security Notification Service is used to keep up with new security updates as they are released.

Intrusion Detection System

I've chose a system called **ICEpac Security Suite**, by the Network Ice Corporation. It is a comprehensive intrusion detection and protection for the entire GIAC enterprise. It may be a little overkill, but it allows room for growth, should GIAC grow to more than one location. I chose it in addition to the syslog data that is sent to the syslog station from the router, and firewall. The Network Ice solution is proactive in the event a GIAC administrator does not catch intrusions reported by the syslogs. It works by way of agents located on each server and workstation ("Black Ice Defenders") as well as network segment monitoring for the DMZ, VPN and internal subnet ("BlackICE Guards").

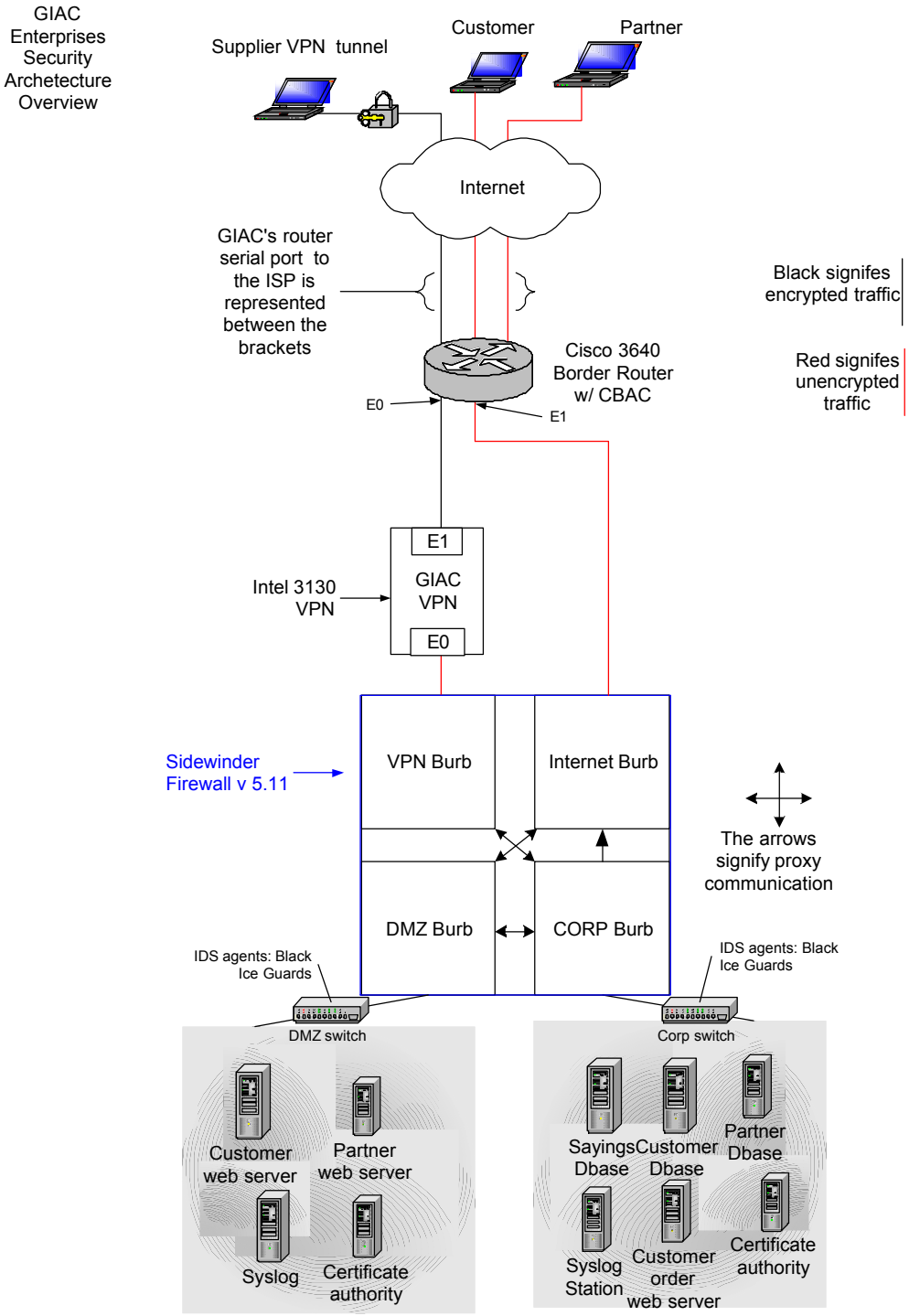


Figure 1

Component Parts of the Network

Border Router

The border router is the first line of defense between GIAC Enterprises Network and the Internet. It is a Cisco 3640 with 2 100mbps fast Ethernet interfaces and 1 high speed serial interface running IOS version 12.1(5)T with the optional Firewall Feature Set (FFS) package. It employs stateful packet filtering. Its primary role is to:

Filter incoming (ingress) traffic from common exploits and the private address space as per RFC 1918. The RFC document states the following IP ranges have been set aside as private IP address space; 10.0.0.0, 172.16.0.0, 192.16.0.0. GIAC is using these private addresses on its internal networks for two reasons. One, they are non-routable, thus more difficult for an attacker to reach from the Internet. Secondly, they are free. We can have as many IP addresses as we need without the expense and trouble of having officially registered IP addresses through Internic. Since these addresses are for internal non-Internet use only, we can be sure that if they approach our network from the outside they must be *spoofed*, (forged) packets and must be denied.

The border router must also provide egress filtering to prevent unauthorized traffic from exiting the premise router without a valid ip address from the internal ip space. This is to prevent ip spoofing and distributed denial of service attacks.

The border router logs information on all packets dropped, in order to ascertain the source of the traffic, so admins can search for possible illegal activity, and errant processes. This is accomplished by sending syslog data to the syslog monitoring station on the internal subnet.

The filtering described above reduces the load on the main firewall, which helps its performance. In addition, the border router performs as a firewall due to its Firewall Feature Set, known as CBAC. Details on CBAC and the border routers' specific Security Policy and access lists are shown in Section II, Security Policy.

Primary Firewall

The primary firewall is made by Secure Computing, and is called Sidewinder. http://www.securecomputing.com/. It is an application layer type firewall. It is the centerpiece of GIAC's Security Policy. All network traffic into or out of GIAC's DMZ and GIAC internal LAN is checked, inspected and proxied before it is allowed to pass. This is true whether it be customers coming from the Internet or suppliers using the VPN. The Sidewinder in my configuration has four burbs. Each burb is a separate unit with its own TCP/IP stack and Network interface. Refer to Figures 1 and 2.

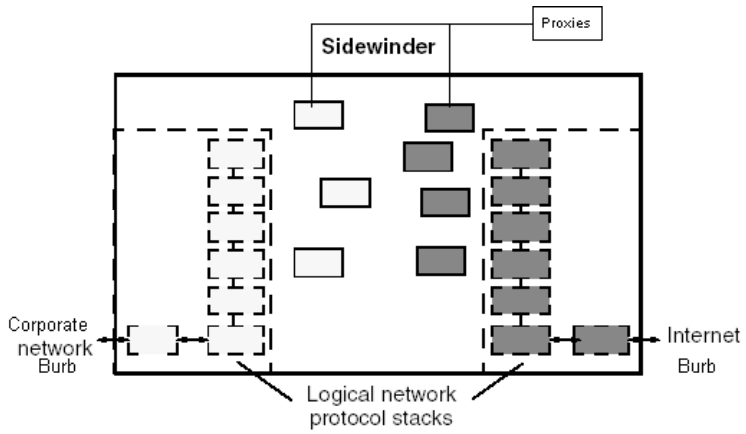


Figure 2

(Image courtesy of Secure Computing)

The only way data can pass between burbs is through proxies and access control lists (acls). Proxies are the go-betweens that permit communications between the different burbs on the Sidewinder. For example, when a user on the internal GIAC internal burb tries to establish an Internet web connection, the Sidewinder intercepts the connection attempt, then queries its acl database for a match. If an acl can be found that permits the action, Sidewinder will open the connection on the user's behalf and pass the data with a web proxy. All Internet connections are made by the Sidewinder so the internal network users and DMZ servers never communicate directly with the Internet. These proxies are set up transparently, so the internal users do not realize that the proxying is taking place.

The Internet burb filters traffic from the Internet. There is a VPN burb placed inside of the VPN device so that it can filter traffic after it is decrypted. There is a DMZ burb that protects the public web servers. The public web servers communicate to the order database on the GIAC internal subnet via an encrypted 128-bit SSL proxy. Authorized traffic is passed by proxy from the VPN burb to the sayings database on the GIAC internal subnet. The Sidewinder's access control lists are found in Section II of this document. One might be concerned as to whether the Sidewinder can keep up with the traffic without becoming a major performance bottleneck. The four-burb design is more resource intensive than a traditional two-burb design. For that reason I chose the Compaq ProLiant DL380 hardware platform to run Sidewinder. It has two 1GHz Intel Processors, two Compaq 18.2 GB disk drives that spin at 15,000 RPM, and 1 GB of ram memory.

VPN Gateway

The Intel Netstructure 3130 VPN Gateway is also a robust unit. It is based on Isolation Systems and Shiva Technology. It has an encrypted throughput of 95Mbps, and is capable of 10,000 simultaneous tunnels, and sports a 733 MHz Pentium® III processor with 512 MB of memory. I like working with it. It uses both a command line interface, which is very similar to Cisco's, and a user friendly GUI. It supports both proprietary SST

tunneling and IPSEC. I've chosen to use SST (Shiva Secure Tunneling protocol). It has proven to be reliable through personal experience. Although IPSEC tunneling is becoming popular and is the defacto standard, there are still inoperability issues between vendors. If I had to use gateway-to-gateway tunnels, I would use the same equipment at both ends. If that were not possible, then I would select the IPSEC protocol.

The GIAC VPN gateway is configured with a number of individual client-to-gateway tunnels, one for each fortune saying author/supplier.

What is a VPN gateway, anyway?

VPN stands for Virtual Private Networking. The VPN concept is to allow secure PC to LAN (local area network), or LAN-to-LAN connections over the Internet without the expense of fixed leased lines. A VPN gateway is usually sold as a standalone device. On the other hand, Microsoft 2000 server and others have the ability to serve as a VPN gateway. Typically, businesses choose standalone solutions for reasons of security, reliability, performance, and support. Physically, a VPN gateway is little more than a PC with two network interface cards (NICS), and an encryptor card. Logically, a VPN gateway encrypts data packets from its trusted side NIC, routes traffic like a router, then it decrypts at the destination gateway. The entire packet including the TCP/IP header is encrypted to provide a virtually impenetrable tunnel of data across the Internet. The VPN Security Policy in Section II tells how I set this up.

GIAC Internal LAN

The GIAC internal LAN is only accessible by GIAC employees. It contains a GIAC internal Intranet, which offers its employees email, and access to the company databases. The order database is where the orders are placed and processed. The orders come in from the public web server located on the DMZ through a firewall proxy, and communicate to the database via a secure SSL connection. In this manner no traffic from the Internet or VPN directly enters the GIAC internal subnet. The GIAC internal LAN has an internal mail server that sends and retrieves email to the Internet through the Sidewinder's secure Sendmail email relay system. Employees process the sayings from the supplier database and add them to the order database.

DMZ

The DMZ is protected behind the DMZ burb of the Sidewinder. It contains web servers for taking orders from Internet customers and GIAC Partners and a certificate authority server for VPN digital certificates and SSL certificates.

Section 2

Security Policy

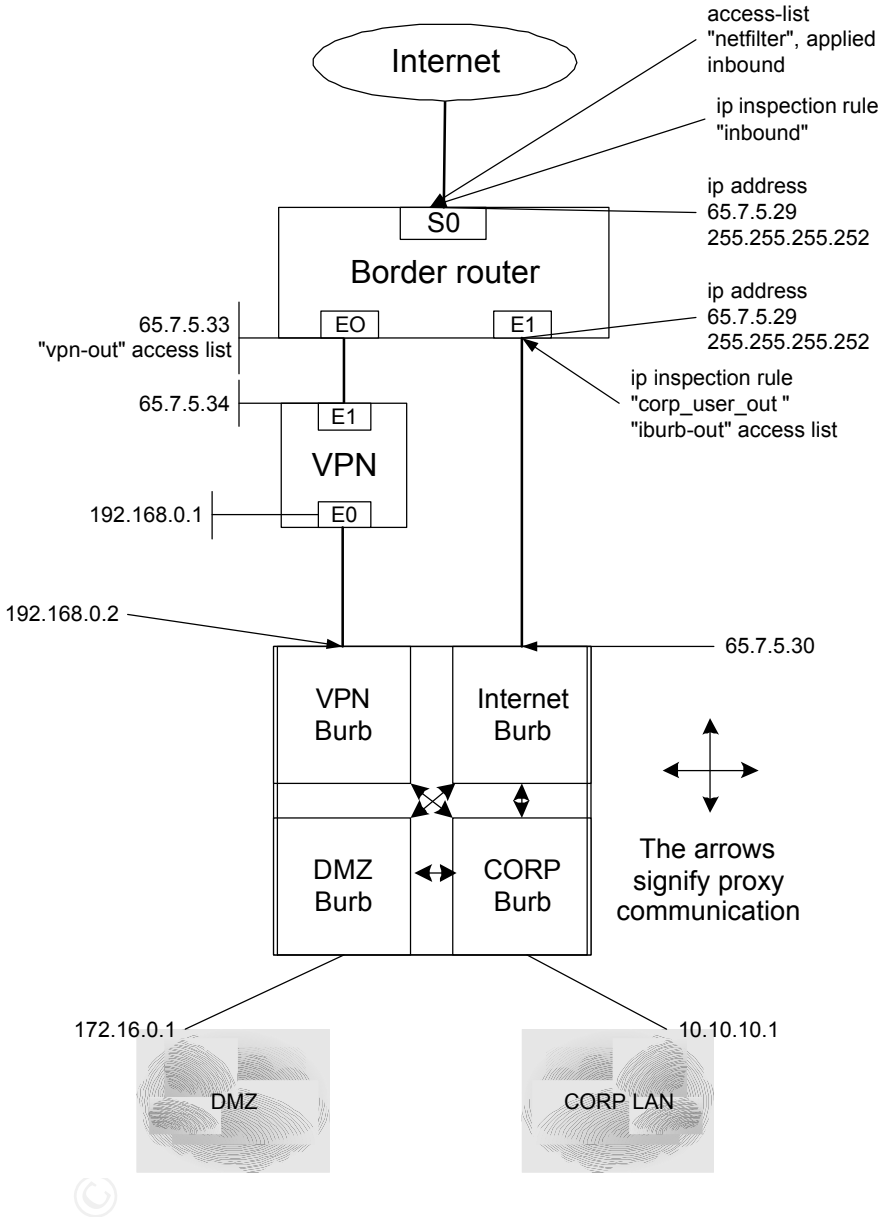


Figure 3

General GIAC Security Policy

Internal GIAC internal users have Internet web access. They may also view Real Media streaming video for watching newscasts only. They can send and receive email, and

download files. ICMP is denied to all, including administrators outgoing and incoming, unless needed. At that time it may be temporarily turned on, then turned off when not expressly needed. Modems are not permitted anywhere on the GIAC premises. VPN access is provided for Suppliers and authorized administrative personnel. Web servers are provided for customers and partners to order products via web based applications designed for that purpose. Customers and partners each visit different web sites to place their orders. Suppliers add their fortune sayings to a custom-written web server application on the GIAC internal subnet. No one from the Internet, DMZ or the VPN is allowed into the internal subnet with the exception of SSL traffic. SSL traffic is authenticated with a certificate authority on using digital certificates. SSL traffic carries web server data and queries between the internal, DMZ, and VPN, segments. Security Awareness Training and Enforcement (SATE) shall be given on a regular basis as determined by the System Security Administrator. Logs are to be kept and reviewed daily of network activity on the border router and firewall. Passwords are to be at least eight characters in length, consisting of uppercase and lowercase letters, special characters and numerals. A tool called Password Policy Enforcer, (PPE) http://www.tpis.com.au/products/ppe/ checks the passwords, to make sure they meet the standard.

This policy is detailed in the pages that follow covering the border router, the Sidewinder firewall and the Intel VPN gateway.

Border Router

The Cisco 3640's security policy permits dns, ftp, http, https, real-audio, and smtp. Permits access to the VPN on port 2233 only. All traffic not expressly permitted is denied.

Access List Background Information

Cisco has several different types of access lists that can be used for access control. Standard access lists are limited in filtering only the source address. Extended access lists filter source and destination addresses, in addition to protocol source and destination ports.

Named access lists are a form or type of extended access list that add the convenience of a user defined name, such as "inbound", or "outbound". Named access lists also enable you to selectively delete specific entries with the "no" command. For example, to remove this permit statement,
permit any host xxx.xxx.xxx, eg smtp, add "no" in front of it
no permit any host xxx.xxx.xxx eg smtp.

Dynamic access lists create specific temporary openings on the fly. An internal user can open a telnet session to a remote host on port 23. That host will respond on a port above 1024, say 1025. The router will then temporarily open port 1025 for that specific host and snap it closed after the session is terminated.

Dynamic lists have limitations. They require static ip addresses. They also require the user to log in to the router first. Not practical for most situations.

Reflexive access lists do not have the limitations of dynamic access lists. They also create temporary openings for ip traffic based on sessions originating from the trusted side of the router. The source, destination port, and ip address are *reflected* back as a mirror image to create the temporary opening. Hence the name, reflexive.

One of the limitations of reflexive access lists is their lack of support for multi-channel applications, such as FTP and streaming multi-media. For more detail on reflexive access lists, visit this link to Cisco's web site.

Now that I have given a little background, I will explain why I chose to use Cisco's strongest form of security outside their Pix firewall. It is called **Content Based Access Control (CBAC).**

CBAC, also known as the Firewall Feature Set, (FFS) supports multi-channel applications such as FTP, Cu-SeeMe, H.323, etc. But, it is much more than that; it is a full security tool kit with the following features:

- Traffic filtering
- Traffic inspection
- Alerts and audit trails
- Intrusion detection

Without CBAC, Cisco routers are limited to inspecting traffic at the network and transport layer. CBAC is smart enough to filter traffic at the application layer, similar to an application layer firewall. It does not inspect application layer information on all protocols, though, only those listed below, and those you define, are available for application level inspection.

• CU-SeeMe (only the White Pine version)
• FTP
• H.323 (such as NetMeeting, ProShare)
• HTTP (Java blocking)
• Microsoft NetShow
• UNIX R-commands (such as rlogin, rexec, and rsh)
• RealAudio
• RTSP (Real Time Streaming Protocol)
• RPC (Sun RPC, not DCE RPC)
• SMTP
• SQL*Net
• StreamWorks
• TFTP
• VDOLive

"When a protocol is configured for CBAC, that protocol traffic is inspected, state information is maintained, and in general, packets are allowed back through the firewall only if they belong to a permissible session". Source Cisco

CBAC inspects traffic traversing the router and builds a state table. This state information is used to create the temporary openings to allow return traffic to a user initiated session.

Application level filtering combined with state table data and inspection rules enables CBAC to detect and prevent against SYN-Flood denial of service attacks.

CBAC will intelligently shut down half-open connections caused by SYN-Floods based on number, time and rate of sessions. The administrator has full control in setting the parameters as to when CBAC will go in and take such action. Table 17 shows the default timeout and threshold values. As you can see by the following table, CBAC is intelligent enough to go in and terminate sessions, thus stopping a denial of service attack, based on the following timeouts and thresholds. These are the default values. The administrator can change the values as he sees fit. (Which I did)

Timeout or Threshold Value	Command	Default
The length of time the software waits for a TCP session to reach the established state before dropping the session.	ip inspect tcp synwait-time (seconds)	30 seconds
The length of time a TCP session will still be managed after the firewall detects a FIN-exchange.	ip inspect tcp finwait-time (seconds)	5 seconds
The length of time a TCP session will still be managed after no activity (the TCP idle timeout).	ip inspect tcp idle-time (seconds)	3600 seconds (one hour)
The length of time a UDP session will still be managed after no activity (the UDP idle timeout).	ip inspect udp idle-time (seconds)	30 seconds
The length of time a DNS name lookup session will still be managed after no activity.	ip inspect dns-timeout (seconds)	5 seconds
The number of existing half-open sessions that will cause the software to start deleting half-open sessions.	ip inspect max-incomplete high (number)	500 existing half-open sessions
The number of existing half-open sessions that will cause the software to stop deleting half-open sessions.	ip inspect max-incomplete low (number)	400 existing half-open sessions

The rate of new sessions that will cause the software to start deleting half-open sessions.	ip inspect one-minute high (number)	500 half-open sessions per minute
The rate of new sessions that will cause the software to stop deleting half-open sessions.	ip inspect one-minute low (number)	400 half-open sessions per minute
The number of existing half-open TCP sessions with the same destination host address that will cause the software to start dropping half-open sessions to the same destination host address.	ip inspect tcp max-incomplete host (number) block-time (minutes)	50 existing half-open TCP sessions; 0 minutes

Table 17 courtesy of Cisco

CBAC also does real time alerts and audit trails, configurable by the administrator. Intrusion detection is also available for a limited number of attack signatures. All these features come at a price, however. Memory and system resources are used in relation to how many protocols or attack signatures are inspected. I chose not to use the intrusion detection for this reason. For more information on Cisco's IDS capabilities view this link at Cisco's web site.

What follows is the configuration I made and my comments about what each line in the configuration means. This configuration has:

- Three extended named access lists.
  - "Netfilter" filters incoming traffic
  - "vpn-out" filters the VPN burb's outbound traffic
  - "Iburb" filters the Internet burb's outbound traffic
- A set of timeouts and thresholds to help CBAC prevent denial of service attacks.
- Two traffic Inspection Rules

Border router configuration

The border router is Cisco 3640 with 64mb of ram memory, and 64mb of flash memory. It runs the Cisco Internetworking Operating System (IOS) version 12.2(3a). It's the newest version I could find with CBAC.

After the router was given a host name and had IP addresses added to it's interfaces, it's time to lock it down to meet GIAC's security policy.
From the user mode prompt, which looks like this: GIAC> We type "enable" then <ENTER>, and the prompt will change to: GIAC#

Then we type in "config t" (meaning configure terminal) then "Enter", and the prompt will change to: GIAC(config)#
This is global configuration mode. Global configuration commands are entered from this

prompt. We start by turning off unneeded services and features. Each command is entered, followed by a press of the "Enter" key after each command. The following commands turn off features that are more useful to attackers than network administrators.

no ip directed-broadcast
no ip source-route
no icmp redirects
no service tcp-small-servers
(small servers are tcp/udp ports 0-20 that run services, such as echo, and chargen. They are disabled because they can be used in denial of service attacks.)
no service finger
no cdp running
no ip http server
(Cisco's IOS has web interface capability but it is known to be vulnerable)
no snmp-server location
no snmp-server contact
(snmp is not needed so it is turned off. Another potential vulnerability)
service timestamps log uptime (how long the router has been up)
service password-encryption (the service password is encrypted)
logging 10.10.10.15 (sends syslog data to this address)

ip subnet-zero (The serial port connection uses the same ip as the fast Ethernet port. Saves ip address space.)

Here is the rest of the configuration;
router rip (a basic dynamic routing protocol)
version 2
network 65.0.0.0
passive-interface Serial 0/0.1 no auto-summary
ip classless (enables use of classless subnet masking)

Now we go into CBAC-specific configration

CBAC configuration

The following are my timeout and threshold values, measured in seconds. They are used to prevent denial of service attacks. *"The timeout value defines the maximum time that a connection for a given protocol can remain active without any traffic passing through the router. When these timeouts are reached, the dynamic ACLs that are inserted to permit the returning traffic are removed, and subsequent packets (possibly even valid ones) are not permitted"*. **Source Cisco**

no ip inspect audit-trail
(it defaults to a global no inspect then the following are added in)

ip inspect tcp synwait-time 30
(the length of time to wait for a TCP session to establish)
ip inspect tcp finwait-time 5
(the length of time TCP is managed after the FIN exchange)
ip inspect tcp idle-time 60 (TCP idle time-out)
ip inspect udp idle-time 30 (UDP idle time-out)
ip inspect dns-timeout 5 (DNS look-ip idle timer)
ip inspect one-minute low 400 (Rate of half-open sessions per minute allowed before CBAC begins closing connections)
ip inspect one-minute high 500
(Rate of half-open sessions per minute before CBAC begins closing connections)
ip inspect max-incomplete low 500
(Maximum number of half-open sessions before CBAC begins closing connections)
ip inspect max-incomplete high 400
(maximum number of half-open sessions CBAC permits)
ip inspect tcp max-incomplete host 50 block-time
(allowed number of half-open sessions with the same destination address before CBAC begins closing connections)

We create inspection rules for the traffic we allow inbound from the Internet to our routers' serial port: These rules allow inbound smtp for our mail server, udp for DNS queries and http and https for access to our web servers. Time out values of 30 seconds each help prevent DoS attacks.

ip inspect name inbound smtp timeout 30 audit trail on
ip inspect name inbound udp timeout 30 alert on
ip inspect name inbound http timeout 30 alert on
ip inspect name inbound https timeout 30 audit trail on

Then we create inspection rules for the traffic we want to allow outbound from the internal users: The internal_user_out rule will be applied against the inbound side of Ethernet 1.

ip inspect name internal_user_out udp timeout 30
ip inspect name internal_user_out tcp timeout 30
ip inspect name internal_user_out smtp timeout 30
ip inspect name internal_user_out http timeout 30
ip inspect name internal_user_out https timeout 30
ip inspect name internal_user_out ftp timeout 30
ip inspect name internal_user_out dns timeout 30
ip inspect name internal_user_out realaudio timeout 30
ip inspect name internal_user_out fragments alert on

Then we create an extended access list inbound on the serial port that faces the ISP. This acl is named “netblock “. It blocks all traffic except rip and packets destined for the VPN

on port 2233. The access list is named netblock. This same access list will be modified in real-time by CBAC's inspection rules to allow return traffic to outbound sessions a way back in.

```
GIAC (config)# interface serial 0/0.1
GIAC (config-if)# ip access-group netblock in

access-list netblock permit udp any 65.7.5.34 eq 2233
! this allows any traffic to reach the VPN on udp port 2233 only

access-list netblock permit udp any eq rip any eq rip
! Rip traffic is allowed for the frame-relay connection

access-list netblock permit tcp any eq http any eq http
access-list netblock permit tcp any eq https any eq https
! web traffic is allowed to reach the firewall, the firewall then filters and proxies   !
web traffic to the appropriate web servers

! We deny any incoming packets matching our own. Any traffic on these ranges is
! spoofed.
access-list 101 deny ip host 0.0.0.0 any log
access-list 101 deny ip 10.0.0.0 0.255.255.255 any log
access-list 101 deny ip 172.16.0.0 0 any log
access-list 101 deny ip 192.168.0.0 0.0.255.255 any log

! We deny Loopback Address
access-list 101 deny ip 127.0.0.0 0.255.255.255 any log

! We deny Multicast Address
access-list 101 deny ip 224.0.0.0 7.255.255.255 any log

! We deny Broadcast Address
access-list 101 deny ip 255.0.0.0 0.255.255.255 any log

! All other tcp packets are denied
access-list netblock deny tcp  any any

! All other udp packets are denied
access-list netblock deny udp  any any

! All icmp packets are denied. CBAC features don't work with icmp
access-list netblock deny icmp any any

access-list netblock deny ip any any
```

! This blanket deny-all statement is invisible, but present at the end of all Cisco ! access lists by default

Now we create a outbound access lists for E0 and E1, They are applied against the E0-E1 interfaces inbound. They control what traffic we will permit outbound from the Internet Burb and VPN burbs.

```
GIAC (config)# interface fast-ethernet 0
GIAC (config-if)# ip access-group vpn-out in
access-list vpn-out permit udp any 65.7.5.34 eq 2233
! all vpn traffic uses this port

GIAC (config)# interface fast-ethernet 1
GIAC (config-if)# ip access-group Iburb-out in
access-list Iburb-out permit ip any 65.7.5.30
! any traffic coming from the Internet burb's IP address is firewall NAT-ed traffic
that has been proxied, so it's ok
```

CBAC in Action

We apply the inspection rule named "internal_user_out" to the E1 interface, inbound. Traffic leaves the internal subnet by entering E1, then exiting S0, and on to the ISP. Therefore, as traffic from the internal subnet enters the E1 interface, it is inspected by the internal_user_out rules, on its way out to the Internet. CBAC records information about the state of any outgoing connections that are initiated by internal users. It adds this connection information to its state table, as the internal user traffic flows through the E1 interface. If this traffic is requesting a web page, for instance, CBAC automatically inserts a temporary acl entry in at the beginning of the netblock access list. This temporary acl permits inbound return packets that are part of the same connection as outbound packets that were just inspected. Because CBAC knows the source, destination ip, and port number when we made the HTTP request, it automatically opens the correct port on our source ip for the return traffic to flow through. When the session is terminated, it immediately shuts down the temporary hole that it created in the netblock acl by deleting the temporary acl entry. This is why we can make such a restrictive acl inbound to our serial port from the ISP. CBAC makes tiny pinpoint openings in the access list as needed to allow return traffic from outgoing sessions to return.

Primary Firewall

Security Policy

How the Sidewinder Firewall Works

As mentioned in Section 1, the firewall permits or denies connections through the use of

access control lists and proxies. When a network request is made, Sidewinder checks ACL entries to determine whether to grant or deny the requested connection. However the ACL database has no effect on the actual flow of packets through the Sidewinder. Network separation has been designed-in by separate TCP/IP protocol stacks as shown in Section 1 figure 2. ACL entries only determine whether the Sidewinder will allow or deny a connection *attempt*. Even if the attempt is made, there must be an associated proxy before the data can flow from one burb to another. Proxies are agents that inspect and pass data at the application layer between the firewall's various burbs.

Type Enforcement

Sidewinder is an application level firewall that runs under SecureOS, a hardened version of the BSD UNIX operating system. Secure Computing has enhanced and with a patented security technology called Type Enforcement. Type Enforcement is based on the security principle of least privilege: any program executing on the system is given only the resources and privileges it needs to accomplish its tasks.

All operating systems (OS) are inherently insecure upon initial set-up. The OS manufacturer builds them to be as easy and convenient as possible to install and configure. Logging in as super-user (root in Unix, or Administrator in NT) gives you access to all system files. An intruder who can acquire root privileges can do anything he wants on a typical Unix or NT system. Additionally, UNIX does not have tight control over how data files are shared among the processes running on a system. So if an intruder managed to break into one area of a system, such as through a Bind/DNS vulnerability, he may be able to gain access to all systems and wreak havoc.

SecureOS mitigates these traditional Unix weaknesses with Type Enforcement (TE). It uses two different UNIX kernels and the concept of domains to diminish any one users' level of access. Each domain has different purposes. Domains are accessed by switching roles, through use of the *srole* command. Once you have switched to a specific system role, your activity is limited to what is permitted in the domain(s) associated with that role. Even if UNIX file privileges indicate that you have permission to use a certain file, Type Enforcement may prevent you from using it. Therefore, user access is determined by the role(s) he has been assigned. A role determines the domains, which the user's programs may operate in. The program's domain determines which files, networks, and system services it may access and what actions can be taken.

The Operational Kernel is the normal operating state for the Sidewinder. In this mode, the Sidewinder is connected to the Internet and to internal networks. All network services are operational.

The Administrative Kernel is only used when an administrator needs to perform administration tasks on the Sidewinder, such as installing software or performing backups. When the Administrative kernel is running, all network connections are disabled. Internet services are not available.

What is a burb?

A burb is a type enforced network area used to isolate network interfaces and their traffic from each other. Sidewinder is capable of supporting 9 burbs. I've chosen to use Sidewinder with four burbs, one for the Internet, one for the VPN, one for the internal corporate users, and one for a protected DMZ.

Sidewinder Firewall Configuration

The access control list shown in Table 1 is designed to meet GIAC's Security Policy. This table is the expression of that security policy. This acl is read from top to bottom by the incoming packets. When a packet finds a match it stops at that point and follows the rule. It goes through all the rules if necessary until it finds a match. If it does not find a match, the packet is dropped (denied).

Sidewinder has a feature called Service Groups. Creating a Service Group allows us to group proxies together. This enables us to create far fewer access control lists. Fewer lists mean better performance and easier management of the firewall.

I created the following Service Groups, and grouped their proxies as shown below:

- Admins
  Telnet, ftp http & https, ports 2233, 10025-10028 for VPN Management
- Giac_Users
  http, https, realmedia, ftp
- Internet-In
  http and https
- VPN_Users
  http and https

After logging in with Sidewinder's remote client, Cobra, I created the Service Groups. I selected Policy Configuration… then "Service Groups". See Figure 4.

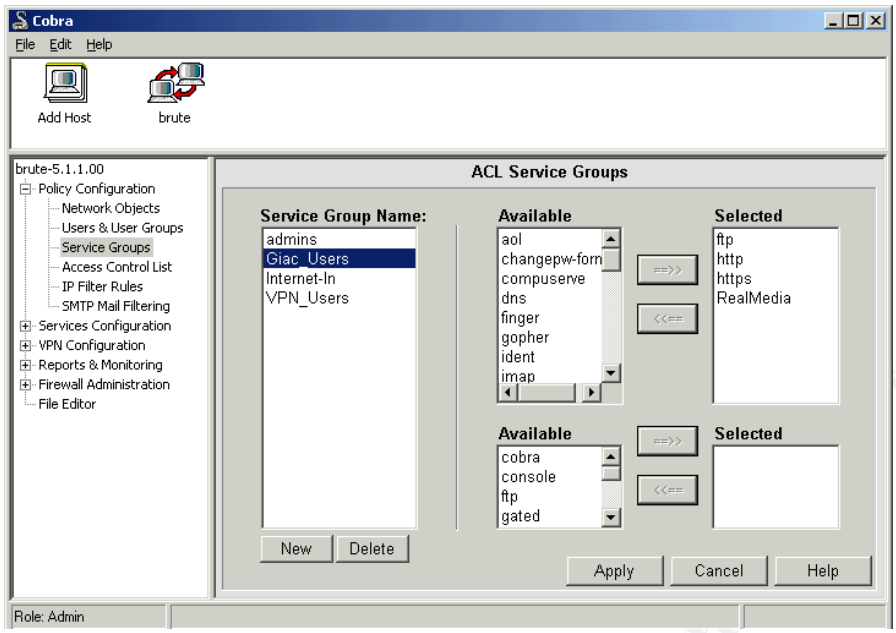


Figure 4

From the Available Proxies list I selected the proxies I needed for each group. Figure 4 shows the proxies chosen for the Giac_Users group.

Figure 5 shows the proxies I chose for the Internet-In Service Group. I followed the same process for the VPN_Users and admins Service Groups. To the admin group additional proxy services were made available for managing the VPN gateway and other administrator tasks. See Figure 6 for the admins service group.

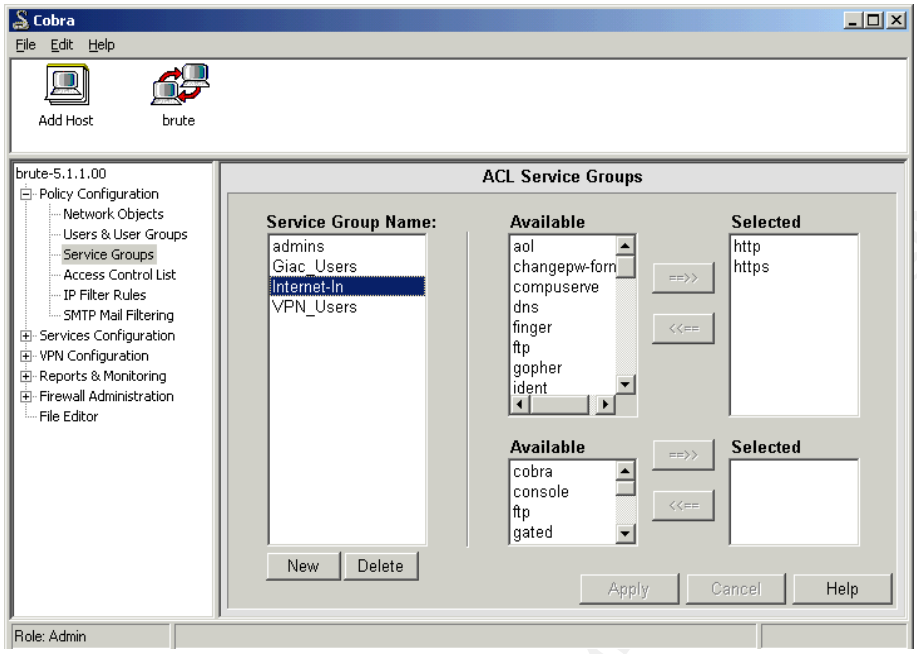


Figure 5

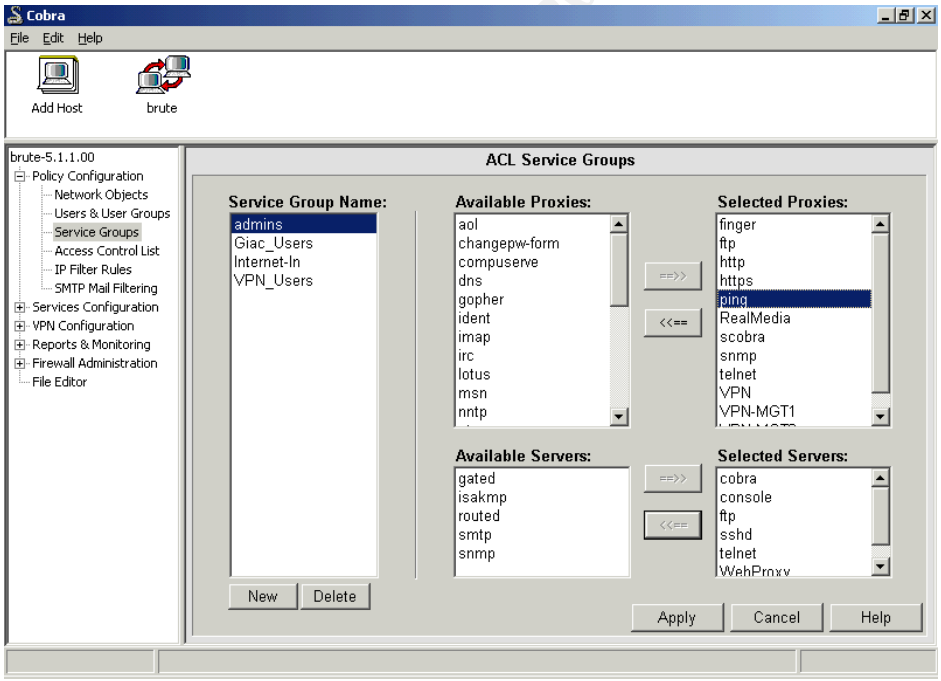


Figure 6

DNS and SMTP (mail) traffic are handled automatically since I configured Sidewinder's split-dns servers and dual Sendmail servers as described in the next section. Thus acl

entries were not needed for them. After creating the Service Groups and proxies I created the access control lists shown in Table 1.

Table 1

Rule #	ACL Name	Source Burb	Dest Burb	Service or Proxy	Summary	Reason
1	Deny_all	All	All	none	Denies all connections from all burbs	We start here to ensure there are no default open "holes"
2	Admin-access	Internal	Firewall	Secure Cobra	Allows remote admin access from INTERNAL burb only via secured SSL client	Admin access from the trusted side only
3	Login-console	Firewall	Firewall	none	Allows logins to the Sidewinder system console	Direct Physical access
4	Customers-in	Internet	DMZ	Service Group: Internet-In	Allows customers and partners coming in from the Internet to access the GIAC public Web site.	Customers & Partners place orders
5	DMZ-Dbase	DMZ	Corp	https proxy	Allows database updates and queries from the DMZ to the corporate databases	Order entry from public web servers to Corporate databases
6	VPN-User	VPN	Corp	Service Group: VPN-User	Allows users coming from the VPN burb to access the supplier Web site on the Corporate LAN.	Supplier access to sayings database
6	VPN_MGT	Corp	VPN	Service Group: Admins	Allows users in the Internal burb to manage the VPN gateway	Manage-ment access to VPN
7	Syslog	Internet	Corp	Service Group: Admins	Allow syslog traffic from the border router to reach the corp burb	Syslog monitoring
7	GIAC-Out	Corp	Internet	Service Group: Giac_Users	Allows GIAC corporate users Internet access as per GIAC Security Policy	GIAC Internet Access
8	Deny-all	All	All	none	Denies all connections from all burbs	Deny and Drop everything else

All traffic is logged by Sidewinder using Unix syslog, and sent to syslog station on the Corp LAN. Sidewinder also provides alerts and SQL audit reports.

DNS

DNS domain name service is used to resolve host names to ip addresses. DNS uses look-up tables to do the resolution. It is a good practice is to have two DNS tables, known as DNS name servers. One is for address resolution of external Internet sites. The other is used for internal address resolution. This is known as split DNS. Attackers like to begin their reconnaissance by reading your internal DNS records, known as a zone file. If an attacker is able to read your internal zone file he can map your internal network and have an easier time in orchestrating an attack against it. Not a good thing. With split DNS, internal users forward DNS queries to the external DNS, and use the internal DNS for everything else. Thus the external DNS server, even if it is compromised, will not divulge internal address information, because it does not have it to begin with. The external DNS is also subject to cache poisoning attacks, in which phony ip address information is inserted which allows an attacker to re-direct an internal user to a phony site, disguised as a legitimate one, for instance a bank. The attacker could use this fake bank storefront to capture credit card numbers and passwords. Another bad thing. So protecting DNS is very important. One must prevent anyone, including your external DNS server, to read your internal zone file except internal users.

Sidewinder has built in features to secure DNS. I would configure it to host split dns on the firewall itself. This way both the internal and external DNS name servers are protected from attack by the firewall. I configured the external DNS server to be a slave to the ISP's DNS server. The internal DNS was configured to be the Master, although I could have configured it to be slave to an existing internal DNS server. The following graphic, courtesy of Secure Computing, shows how this works. If an internal user needs DNS resolution for an external site, the DNS query is forwarded to the external DNS server, which in turn forwards it to the ISP's name server, and so on until the resolution information can be found.

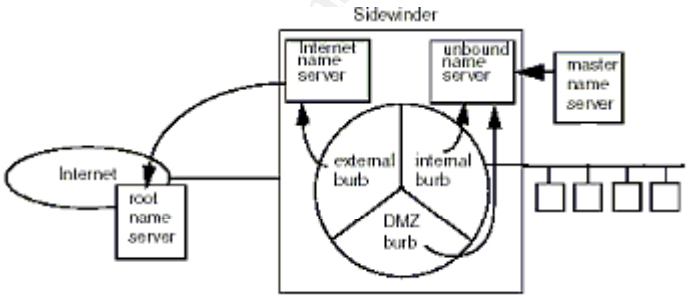
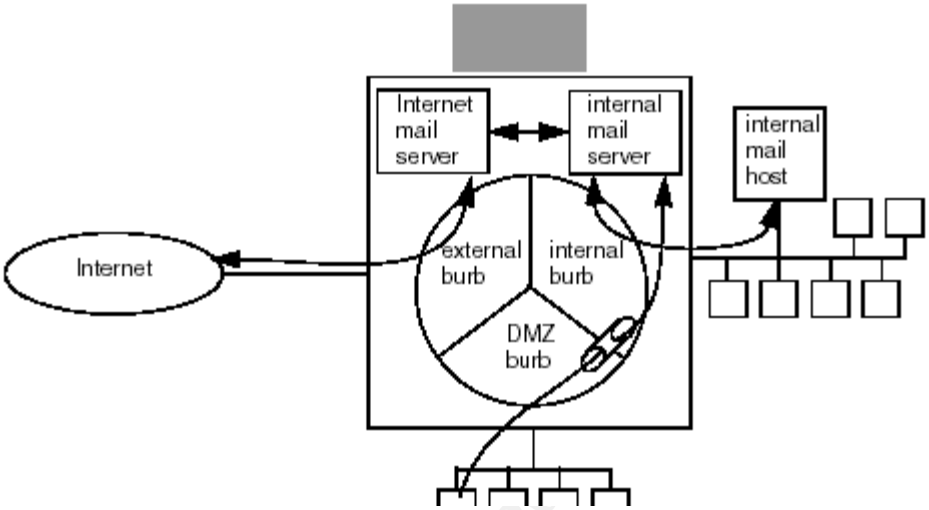


Figure 7

Graphic courtesy of Secure Computing

Securing EMAIL

The next area to be covered is email. Obviously, email is very important to any person or business on the Internet. There have been many well-known attacks and viruses using email, such as Melissa or the I Love You virus. For this reason, Security Awareness Training and Enforcement (SATE) is a vital part of any good security policy. It's important to note that an administrator who builds a secure network but does not have regular, attention-getting SATE training is leaving himself wide open to email attacks and social engineering. Virus protection updates and timely emails from the administrator to warn end users about current threats are very important.



8

Graphic courtesy of Secure Computing

GIAC's Sidewinder uses dual email servers hosted on the firewall to secure email. It works similar to split dns. Incoming Internet mail reaches the external mail relay and is forwarded to the internal mail relay, which in turn forwards the mail onto the GIAC Microsoft Exchange mail server. Outgoing mail is sent from Exchange, to the internal mail relay then to the external mail relay and on to the Internet according to the MX records. Both DNS and mail servers have been hardened by Secure Computing and use Type Enforcement as well for the utmost in secure operation

VPN Security Policy

An Intel 3130 VPN Gateway is used as a portal to the GIAC's web servers for suppliers to use. Suppliers, (fortune authors) are authenticated through a client-to-gateway tunnel then filtered at the firewalls' VPN burb, which allows only VPN udp port 2233 traffic. From the VPN burb, the supplier's decrypted traffic is limited to accessing a web server on the internal corporate subnet. This communication is through firewall's http and https proxies. The https proxy allows an SSL tunnel from the supplier's web browser to the web server, and no more. The web server in turn queries and updates the fortune cookie sayings database, which is also located on the internal subnet.

Customers and Partners place orders public web servers located on the DMZ. If partners required more access than simply placing orders, I would grant it and secure it through the VPN as well. If partner's traffic continued to grow, I would establish a gateway-to-gateway VPN tunnel with them for greater speed and security. The gateway would still terminate at the VPN burb. I would then have a Partner LAN to keep Partner traffic segmented away from the internal corporate network.

VPN Security

VPN technology is fairly secure; however it does have some weaknesses. The biggest weakness is at the beginning and end of the VPN tunnel, before and after traffic is encrypted and decrypted respectively. Measures must be taken to mitigate those weaknesses. The VPN burb filters and logs decrypted traffic before allowing it to proceed anywhere. Then proxies strictly control it. This mitigates the risk of trusted VPN users who have been compromised by an attacker.

GIAC requires its VPN users to sign a document promising not to attack or use gross negligence in allowing an attack on GIAC lest they be held liable.

The agreement also specifies the VPN end user must use an approved personal firewall, such as Black Ice or Zone Alarm. Each end user must have an up to date virus-scanning program with a contract to keep up with the latest virus updates. GIAC configures the VPN gateway to force the client to use a 0.0.0.0 0.0.0.0 default route. This route forces all traffic through the VPN tunnel and denies all other traffic. In this manner a VPN user cannot surf the web and use the VPN tunnel at the same time. Once the tunnel is established, all other traffic is blocked This is how we mitigate the risk of an attacker infiltrating a trusted machine and enter the GIAC VPN tunnel.

All VPN access is over the Internet. A user establishes an Internet connection with his ISP then starts the client software to open a VPN tunnel session with GIAC. Direct dial modem sessions to GIAC are not supported. This forces all traffic to go through the Sidewinder firewall, and mitigates attacks from dial up users, because modems are not permitted anywhere at GIAC.

VPN Policy Rules

I chose to use SST tunnels, (Shiva Smart Tunneling) for the encapsulation type, rather than IPSEC. It uses many of the processes of IPSEC and it is proven to be reliable by personal experience. Although IPSEC is fast becoming the defacto standard in VPN encapsulation, it is relatively new, and so there are interoperability problems between manufacturers. It is best to use the same vendor if you want to build gateway-to-gateway tunnels.

GIAC uses two authentication methods, certificates, and challenge phrases. The public key length I used is 2048 bits. Public keys are used during the authentication and session key exchange processes. The longer the public key length, the more secure the session

negotiation will be. 2048 bits is the maximum key length the Intel VPN offers. The Intel VPN uses the Diffie-Hellman key exchange protocol.

The crypto period length defines how long a session key will be used. The default value for the crypto period is 1 month; we always set it to 24 hours. Intel states *"a packet encrypted with a 90-bit key will require about 20 years of effort by a well-funded dedicated, adversary to crack"* (April 2001, Intel® NetStructure™ Virtual Private Networking Concepts Guide, pp 40).

How SST (Shiva Smart Tunneling) works

SST encrypts each packet of data with a unique packet key. The encrypted packet encapsulates the entire original packet including the header. The original source and destination addresses, protocols and ports are thereby concealed from view. The packet key used to encrypt the packet is appended to the new packet. This new packet is then encrypted with a session key using 3DES symmetric encryption algorithms. 3DES uses three operations with three keys to provide an effective key length of 168 bits. Both the encryptor, (source) and decryptor, (destination) each have a "mathematical half" of each session key. The actual key exchange occurs dynamically using the Diffie-Hellman Authenticated Key Exchange Protocol. (DH) Therefore the session key is never actually sent across the tunnel in the clear, it is created anew for each VPN session and renewed according to the crypto period defined. We use the 2048

3DES RSA key length with a 24-hour crypto period. Thus an attacker has only 24 hours to attempt to decode the 2048 bit encryption. At this time it is mathematically impossible even with clustered computers running at gigahertz speeds.

The following edited syslog text shows the key exchange in action.

```
Negotiating with xxx.xxx.xxx.001
Received negotiation request from xxx.xxx.xxx.002
Replying to negotiation request from xxx.xxx.xxx.002
Generating SST   2048-bit queued DHValue (RSA)
Received negotiation reply from xxx.xxx.xxx.001
Generating SST   2048-bit queued DHValue (RSA)
Requesting key agreement with xxx.xxx.xxx.001
Received key agreement request from xxx.xxx.xxx.001
Session established with xxx.xxx.xxx.001
Generated  SST   2048-bit queued DHValue (RSA)
Received key agreement reply from xxx.xxx.xxx.002
Session established with xxx.xxx.xxx.002
```

VPN Tutorial

What follows is a step-by-step procedure to set-up and configure the Intel 3130 VPN gateway. After that I will connect to the gateway from a laptop with the client software. The Intel 3130 may be configured by command line or through a GUI interface. I use both for speed and accuracy.

Step 1 Establish Physical connectivity

What you need:
A laptop with a known good Ethernet card and dongle
A console cable with 9 pin serial cables at both ends
A crossover CAT-5 Ethernet cable.
Hook up the console cable from COM port 1 of your laptop to the console port of the VPN box. Be careful not to connect it to the async port.
Hook up the crossover cable from your laptop's Ethernet card to the e0 port of the VPN box

Step 2 Establish connectivity

What you need:

I recommend using HyperTerminal Private Edition, and Intel's VPN Manager, version 6.9 or newer. Configure your laptop's Network Neighborhood Properties NIC card to an IP address on the same subnet as the internal interface (E0) of the VPN box.

Open HyperTerminal; name the Session VPN console or something similar.
"Connect Using" Com1. Set the bits per second to 9600. Leave everything else at its default setting.
Hit the return key a couple of times until you get a prompt.

Step 3 Enter Setup Mode

You may get a license agreement screen, if so; agree to it to get back to the prompt.
Type enable, then hit the return key.
Enter the default enable password which is shiva .
Type setup (that will start a series of set-up prompts).
At the "hostname" prompt, type the hostname you want the VPN gateway to have. It's case sensitive.
At the "Bridge mode" prompt, type n (no)
At the E0 IP address prompt, enter the internal IP address of the VPN box (172.16.0.1).
At the E0 subnet mask prompt, enter the internal subnet mask of the VPN box (255.255.255.0).
At the E1 IP address prompt, enter the external IP address of the VPN box (65.7.5.34)
At the E1 subnet mask prompt, enter the internal subnet mask of the VPN box (255.255.255.0).
At the default gateway prompt, enter the IP address of the external router's internal address. (65.7.5.33).

At the VPN manager password prompt, enter the VPN GUI manager password.
Reminder it, you will use it later.
At the time zone prompt, enter MST 7 MDT or whatever your time zone is.
For the hours and minutes use atomic time .
Save the settings when asked.
Now enter the telnet password
Type password from the NORMAL prompt
You will be prompted for the existing password which is shiva
Now enter the new password twice when prompted.
Ping the VPN gateway from your laptop to verify connectivity. Now you will be able to use the TFTP Server and the VPN Manager software through the Ethernet port connection between your laptop and the VPN box. So far all communication has been through the console port.

Step 4 Upgrade the Operating System

The 6.81 OS that the Intel VPN's are shipped with have known connectivity issues. We always upgrade the OS firmware to version 6.9 or newer.
Run the Cisco TFTP Server.
Locate the VPN firmware, version 6.9 or newer.
Under View…Options…Set the TFTP Root Server Directory to wherever the firmware files are. They are named isbr.exe and lrvg.exe.
Issue the command "copy from xxx.xxx.xxx.xxx isbr.exe" Substitute the xxx's with your TFTP servers ip address, it's the same as your laptops' ip address.
Follow it with "copy from xxx.xxx.xxx.xxx lrvg.exe".
After both downloads are completed, reboot the VPN.

The following log file output that shows me connecting to the VPN and doing the initial setup as I just explained in Steps 1-4.

```
hostname:NORMAL#setup
To Exit setup without changing the runtime configuration, press
<Esc> twice now.
Enter Hostname [hostname]:GIAC
Bridge Mode On (Y/N) [n]:
Enter int E 0 IP Address [0.0.0.0]:192.168.0.1
Enter int E 0 Subnet Mask [255.0.0.0]:255.255.255.0
Enter int E 1 IP Address [0.0.0.0]: 65.7.5.34
Enter int E 1 Subnet Mask [255.0.0.0]: 255.255.255.0
Enter Default Gateway [0.0.0.0|interface no]: 65.7.5.33
Enter Manager password [password]:********
Enter time zone [GMT]:MST 7 MDT
Enter year [2001]:
Enter month [9]:
Enter day [17]:
Enter hour [15]:21
Enter minute [8]:
Enter second [42]:
Do you wish to save configuration to flash? [y]:
GIAC:NORMAL#
```

Now I check for connectivity with a ping from the laptop's Ethernet port to the VPN's E0 port.

```
GIAC:NORMAL#ping 192.168.0.1
!!!!!
Success rate 100 Percent (5/5), trip time (min 3 ms, max 4 ms,
average 3 ms)
```

That's good. Now I do a "show directory" command.

```
GIAC:NORMAL#sh dir
Directory listing of *.*

EDI                            0
<VOLUME>
DISKKERN BIN                35188
DH       DAT                 1596
ISBR     EXE              2154848
LRVG     EXE              2117808
IF       CFG                   20
LICENSE  TXT                   11
ISBR     CFG                 2107
BOOT     CFG                  122
Disk Size = 16007168 Free Space = 11595776

Read: 0 Write: 0 Verify: 0 Update: 0
```

LRVG.exe is the Normal Mode executable file, ISBR.exe is the Safe Mode executable file. Using TFTP commands, I copy the new firmware from the laptop to the VPN gateway, one at a time. Then I reboot it.

```
GIAC:NORMAL#copy from 192.168.0.2 isbr.exe
receiving lrvg.exe from 192.168.0.2 via tftp
# blocks transferred
13 103 203 303 403 503 603 703 803 903 1003 1103 1203 1303 1403
150316031703 803 1903 2003 2103 2203 2303 2403 2503 2603 2703
280329033003 3103 3203 3303 3403 3503 3603 3703 3803 3903 4003
41034203 4303 4403 4503 4603
Transfer Completed
No of Blocks transferred is 4702
Transfer Completed

GIAC:NORMAL#copy from 192.168.0.2 lrvg.exe
receiving lrvg.exe from 192.168.0.2 via tftp
# blocks transferred
13 103 203 303 403 503 603 703 803 903 1003 1103 1203 1303 1403
150316031703 803 1903 2003 2103 2203 2303 2403 2503 2603 2703
280329033003 3103 3203 3303 3403 3503 3603 3703 3803 3903 4003
41034203 4303 4403 4503 4603
No of Blocks transfered is 4616
Transfer Completed
GIAC:NORMAL#reboot
please confirm (y/n)y
Rebooting in 2 seconds
```

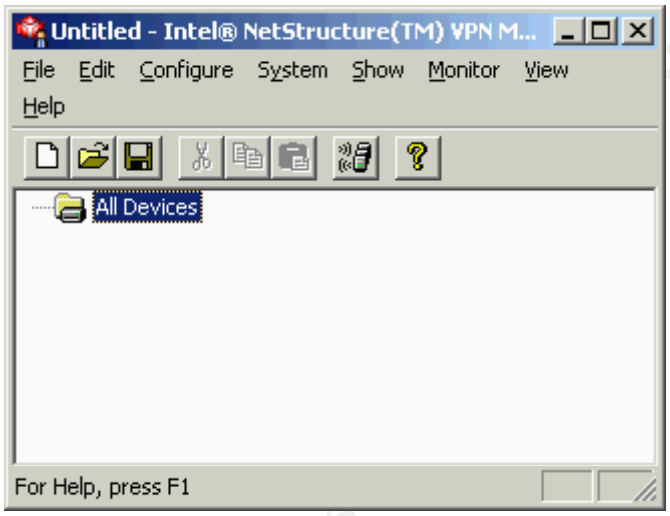
Here are some of the messages as it reboots; This shows the OS upgrade was successful.

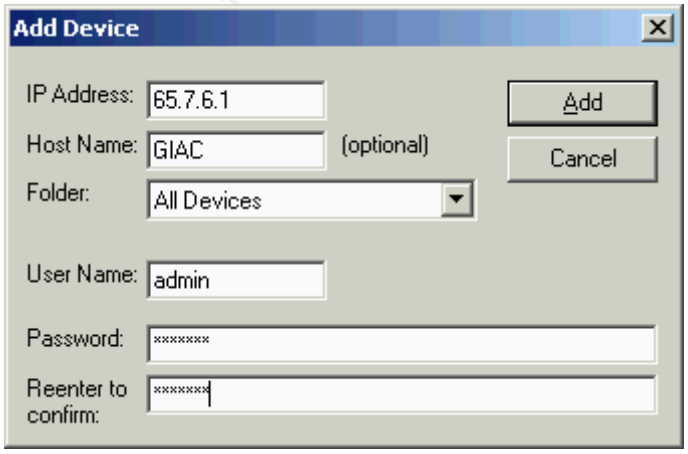
```
Intel(R) NetStructure(TM) 3130 VPN Gateway V6.90
North American Version
Ethernet 0 ha 00 80 D3 F1 97 7A ...hardware up
Ethernet 1 ha 00 80 D3 F1 97 7B ...hardware up
Serial 0 ...hardware up
Async 0 ...hardware up
Encryption device...software up
Random number generator...software up
```

As we continue on to Step 5 in our configuration of the VPN gateway, note the ip addresses used in my screen shots are a little different than the actual ip's used in my design. They still serve for illustrative purposes.

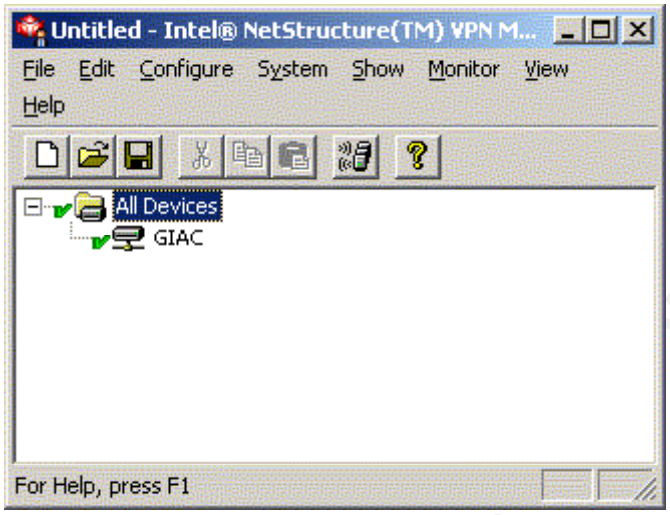
Step 5 Run the VPN Manager (GUI) Software

Its opening screen is shown here.

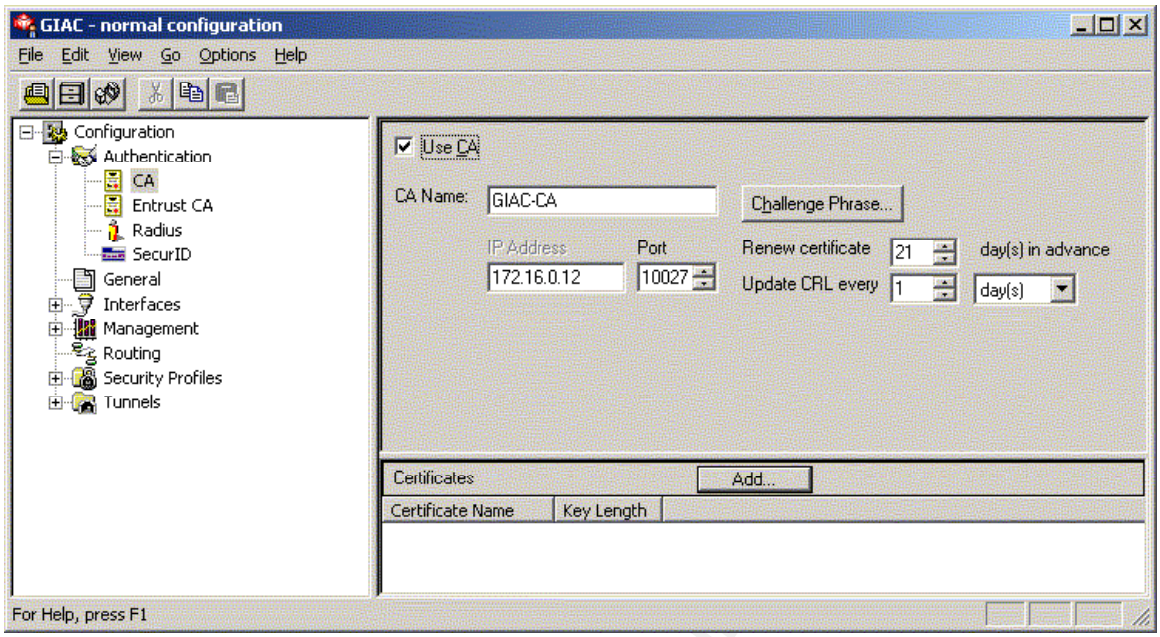


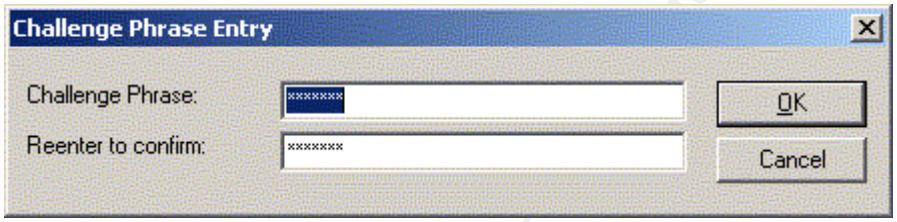


Under the File Menu select "Add New Device"
Enter the VPN's E0 ip address.
Type admin for the User name field
Enter the VPN password twice (same one you entered from the command line earlier)
Leave the other fields blank.
You will see <Unnamed Host> icon on the VPN Manager's screen.
Double-click on it to read the configuration. When you close it, it will display the proper hostname.



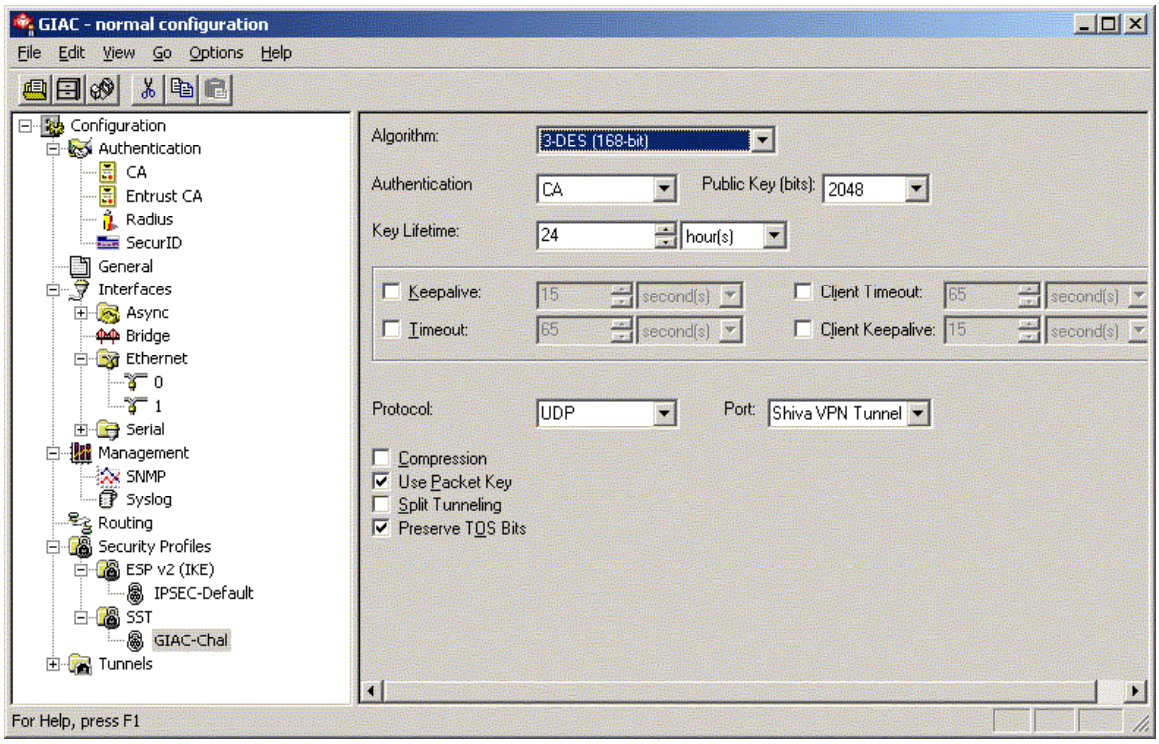
Step 6 Finish building the Configuration with VPN Manager
Define the Certificate Authority (CA)
172.16.0.12
Create a certificate



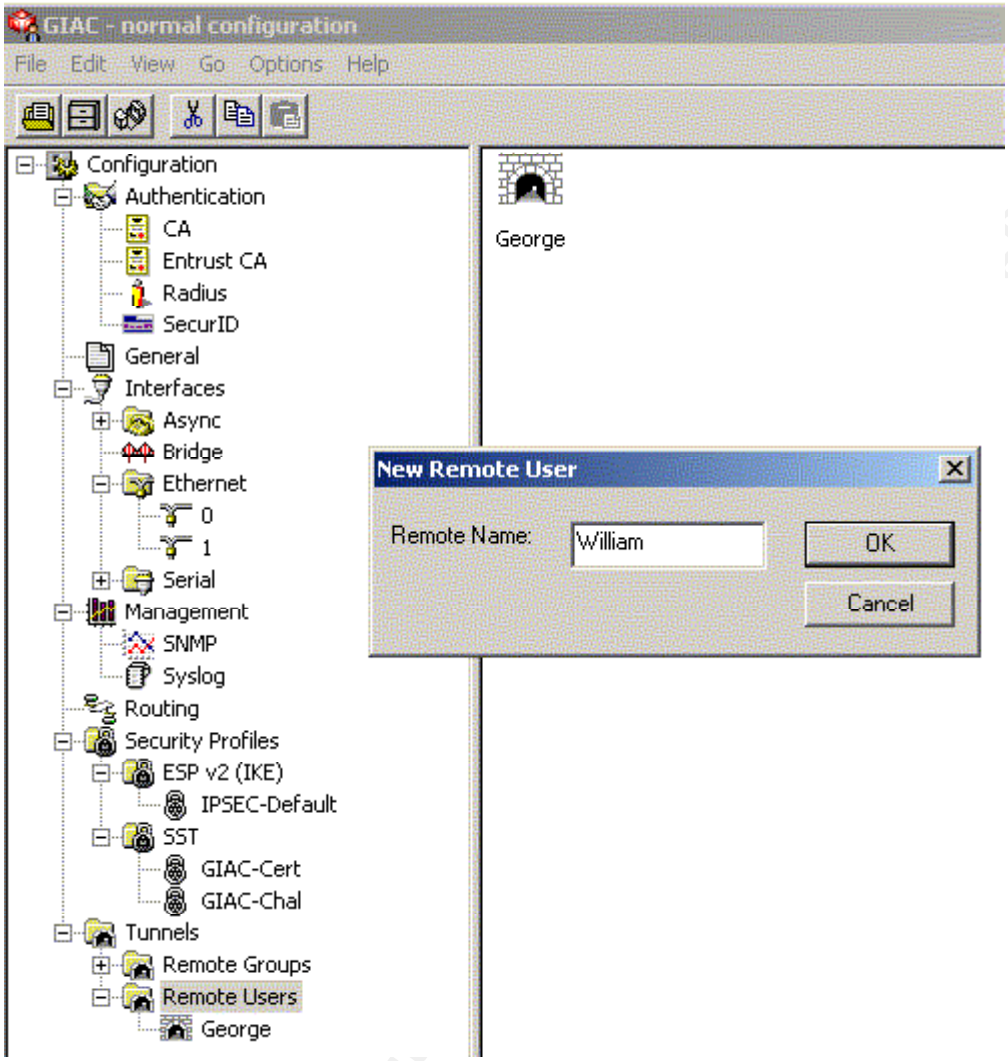


The challenge phrase must match what you've given the remote client

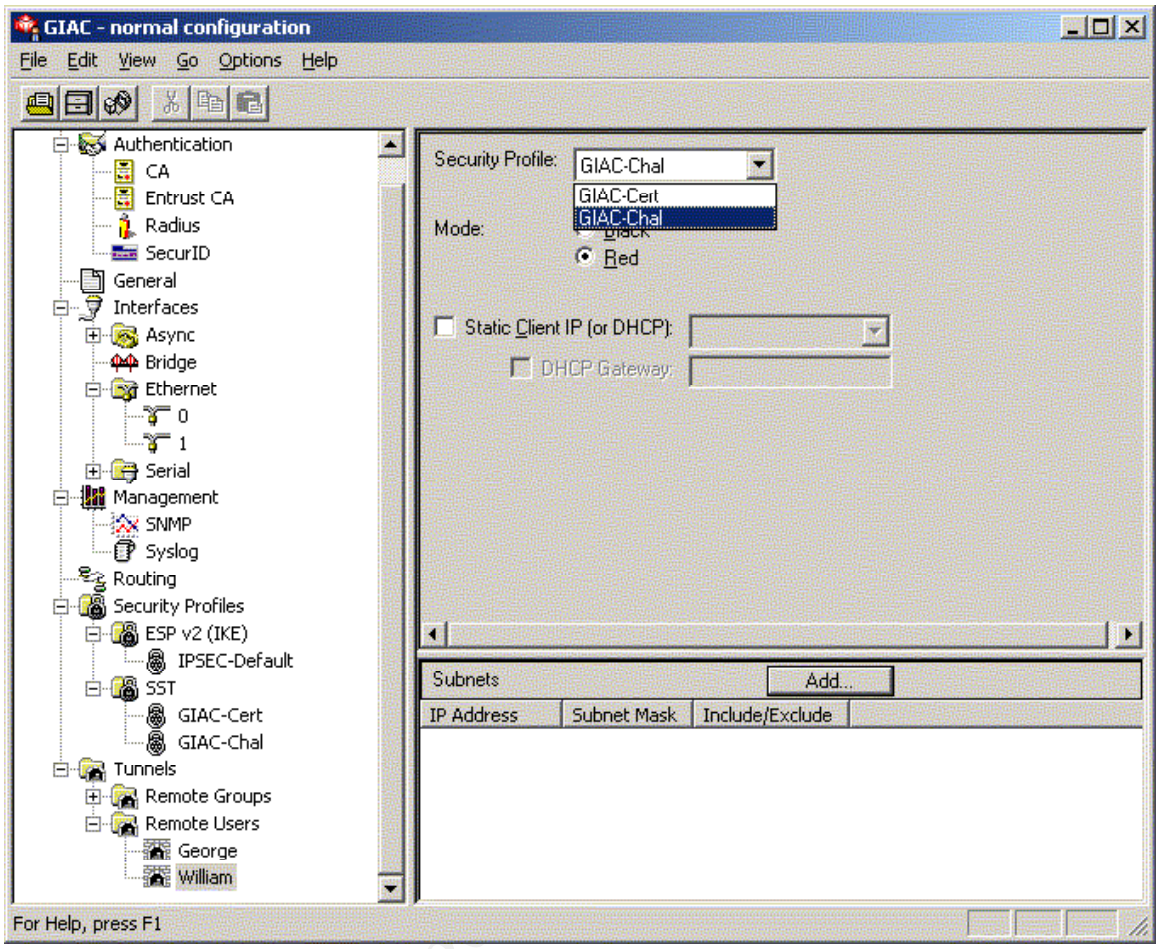
Security Profiles
Right click on Security Profiles….New Security Profile >…SST…"Profile Name", enter GIAC-Chal
Right click on Security Profiles….New Security Profile >…SST…"Profile Name", enter GIAC-Cert
Set it's properties as shown on here. The remote client settings must match or the tunnel will not be established!



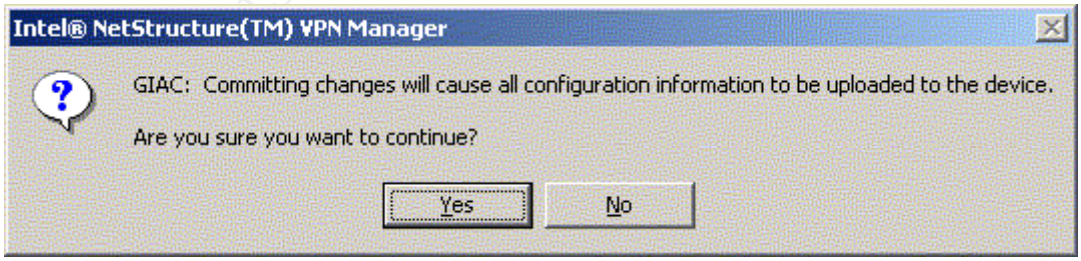
Define your client tunnels
Right click on “Tunnels”….”New Tunnel”…SST> “Remote User > ”.



The tunnel name you create must match the client, and it is case sensitive. Notice in the next screen shot that you can select either the GIAC-Cert Profile, or the GIAC-Chal profile. I always make both a Challenge-Phrase Security profile and a Certificate-Profile. In the event the client cannot establish a tunnel with a certificate, connecting by challenge phrase is a way to isolate the problem and test that everything else is working properly. Certificate authentication is much more secure however, because you have a certified third party (the cert authority) that verifies the authenticity of both the remote client and the gateway. We normally generate a cert on the cert authority server, then save it to a floppy. The cert files are then sent via encrypted PGP email to the customer that is getting the remote tunnel access. He then calls us, after we verify his identity over the phone we tell him what the pass phrase is. We also send him a PowerPoint slideshow that describes on how to install the client.



Next we Setup the Syslog,
Under Management…Syslog, Select Host…We set the Priority to Debug. Click Add Host enter 192.168.0.12. We use syslog data for troubleshooting.



We save the changes and upload them to the VPN gateway. We have now finished building the VPN client tunnel definitions for William and George.

What follows is the GIAC VPN gateway configuration file: My comments inserted inline.

```
!!!!! NORMAL CONFIGURATION
!
hostname GIAC
timezone MST 7 MDT
sntp 0.0.0.0 123 0
!
int e 0 !The e0 interface is the trusted side
     ip address 192.168.0.1 255.255.255.0
     ip mtu 1500
     mode red (mode red  means the tunnel terminates on red)
     bandwidth AUTO
     duplex FULL
     dhcp-relay disable
int e 1
     ip address 65.7.5.34 255.255.255.0
     ip mtu 1500
mode black
! e1 is mode black, the un-trusted side that faces the Internet

bandwidth AUTO
! the bandwidth is set to AUTO so it will sync up with 10 or 100
! megabit networks
     duplex FULL
     dhcp-relay disable

int s 0
!This interface is not needed so it is shutdown!
     shutdown
     ip address 0.0.0.0 0.0.0.0
     encapsulation frame-relay
     ip mtu 1500
     mtu 2048
     mode red
     bandwidth 1150000
     keepalive 0
     frame-n391 6
     frame-n392 3
     frame-n393 4
     frame-lmi lmi
     frame-conform none
     dte
int a 0
!This interface is not needed so it is shutdown !
     shutdown
 ip address 0.0.0.0 0.0.0.0
     encapsulation ppp
     ip mtu 1500
     mtu 2048
     mode red
     bandwidth 115200
     keepalive 0
     chat ""
     idle-timeout 10
     chat-timeout 30
     compression off
     ppp-authentication pap
bridge 0.0.0.0 0.0.0.0 ! bridge mode is not used is this case!
```

```
ip red-gateway  0.0.0.0
! sometimes it's helpful to define two gateways to direct traffic
which  ! way to go
ip black-gateway  0.0.0.0
ip default-gateway  65.7.5.33
!this is our default gateway for this example. It's normally the
! external router's internal interface

secure-profile GIAC-Chal
!This is the Security Profile using a challenge phrase

     encapsulation sst (SST tunnel encapsulation)
     authentication key (authentication type is session key)
     preserve-tos on
     public-key-length 2048 (RSA 2048 bit)
     algorithm 3des (168bit)
     crypto-period 24
      (24hr interval between before renewing the crypto keys)
     timeout 0
     keep-alive 0
     client-timeout 0
     client-keep-alive 0
     compression off
     protocol 17
     packet-key enable
     split-tunnel disable
      (can't surf the web and VPN tunnel at the same time)
secure-profile GIAC-Cert
!This is the Security Profile using a digital certificate
     encapsulation sst (SST tunnel encapsulation)
     authentication certificate (authentication type is certificate)
     preserve-tos on
     public-key-length 2048 (RSA 2048 bit)
     algorithm 3des (168bit)
     crypto-period 24 (24hr interval between renewing the crypto
keys)
     timeout 0
     keep-alive 0
     client-timeout 0
     client-keep-alive 0
     compression off
     protocol 17 (udp)
     packet-key enable
     split-tunnel disable
      (can't surf the web and VPN tunnel at the same time)
remote William (this is the remote tunnel for William)
     tunnel-type sst
     mode red
     profile GIAC-Cert (This is the certificate security profile)
     client-ip 0.0.0.0 (this forces all packets through the tunnel)
remote George (this is the remote tunnel for william)
     tunnel-type sst
     mode red
     profile GIAC-Chal (This is the challenge phrase security
profile)
     client-ip 0.0.0.0 (this forces all packets through the tunnel)
     auth-key
```

```
ca 192.168.0.12 10027 (The certificate authority uses UDP port 10027)
     renew-cert 21 (It renews the certificates every 21 days)
     update-crl 24
      !It updates the certificate revocation list every 24 hours
     caname GIAC-Ca (The certificate server hostname)
     ca-auth-key xxxxxxx

     Secondary Syslog hosts
     syslog host 192.168.0.12 514
     !syslog data is sent to this host via udp 514
     syslog facility 4
     syslog priority debug
!The syslog priority is set to debug which provides the most detail

manager-allow red-interface ( We select allow the VPN Manager GUI
application  only on the red, trusted interface (E0)
manager-protocol 17 (uses UDP port 514)

end

!!!!! BOOT CONFIGURATION

bootimage normal
safe-mode enable
safe-timeout 60
!The VPN manager has a Normal mode and a Safe mode. Safe mode is used
! as a backup. This line says to boot up in Normal mode afer 60
seconds ! in Safe mode)
log-buffer 100 (lines)
console-mode enable
```

The following is what the client sees when he successfully connects and successfully negotiates the vpn tunnel. Read it from the bottom to the top.

```
[6:04:48 PM] [tunnel]:Secure tunnel established with 21x.229.xx.92
10/28/2001
[6:04:48 PM] [tunnel]:Timeout for 21x.229.xx.92 set to 0 seconds
10/28/2001
[6:04:48 PM] [tunnel]:Keepalive for 21x.229.xx.92 set to 0 seconds
10/28/2001
[6:04:48 PM] [tunnel]:Access granted to network 0.0.0.0-0.0.0.0 via
21x.229.xx.92 10/28/2001
[6:04:48 PM] [tunnel]:Received key agreement reply from 21x.229.xx.92
10/28/2001
[6:04:48 PM] [tunnel]:Requesting key agreement with 21x.229.xx.92
10/28/2001
[6:04:48 PM] [tunnel]:Received negotiation reply from 21x.229.xx.92
10/28/2001
[6:04:48 PM] [tunnel]:Negotiating with 21x.229.xx.9210/28/2001
[6:04:48 PM] [tunnel]:Creating DH value 10/28/2001
[6:04:47 PM] [tunnel]:Connecting to 21x.229.xx.9210/28/2001
[6:04:47 PM] [cert]: CRL data loaded from disk10/28/2001
[6:04:47 PM] [cert]: CA current certificate loaded from disk - GIAC-
2048 10/28/2001
[6:04:47 PM] [cert]: My current certificate loaded from disk - GIAC-
```

```
204810/28/2001
[6:04:46 PM] [cert]: CA next certificate data saved to disk - GIAC-
2048 10/28/2001
[6:04:46 PM] [cert]: My next certificate data saved to disk - GIAC-
2048 10/28/2001
[6:04:46 PM] [cert]: CA current certificate data saved to disk - GIAC-
2048
```

Section 3, Audit

Technical audits are useful for a number of reasons. They must be carefully planned and carried out with attention to detail. One reason for an audit would be to survey traffic being used and run on a network in preparation of adding a new firewall to it. It's essential to know what protocols and traffic are being used, so there are no surprises when the firewall is deployed and begins blocking all traffic except what's permitted by the Security Policy. Pre-firewall survey audits presented to management before installation of the firewall can save management, end users and the firewall installer a lot of headaches. For example when certain users complain they can't do what they used to, you can point to the management approved security policy that says AOL Instant Messenger and the downloading of mp3 music is no longer permitted. It's good to get those kinds of potential controversies out in the open up front. A pre-firewall audit may also uncover excessive broadcast traffic or misbehaving network applications that are needlessly consuming bandwidth. Correcting and perhaps redesigning the network for optimum efficiency can prevent performance issues down the road after the firewall is deployed.

Another reason for an audit would be to test your firewall to be sure it permits and denies the traffic and protocols you intend it to, before you hook it up to the network. This will also save you some headaches in the event mail is not flowing. Don't want that to happen during a busy workday!

A third reason for an audit is one of the most important, that is a scheduled security assessment, also known as a vulnerability analysis. A vulnerability analysis can and should be conducted from outside the network as well as internally. Software tools such as Internet Security Scanner (ISS) or Nessus are used to scan and survey the entire network. ISS can be used to generate detailed reports on the vulnerabilities it found and what to do about them.

For the purpose of this assignment, I will present my approach in conducting a technical audit of GIAC's primary firewall, Sidewinder.

Let us assume I have built the firewall software and configured it. Now it is time to begin testing it before it is deployed on the network. I will use four laptops.

I will start by putting one laptop on the Internet burb and one on the VPN burb. Although there are many different auditing tools available. NMAP is probably the best for two simple reasons; it is free, it works! I have used several other port scanners with nice GUI interfaces that just do not find open ports like NMAP does.

Volumes could be written about the exploits that today's networks are subject to. I will keep this audit short and sweet. The goal is to only allow what my firewall rule set is supposed to allow. Any other open ports are subject to exploits by attackers.

A visit to SANS website at http://www.sans.org/top20.htm provides a list of the SANS/FBI Top Twenty Security Vulnerabilities and detail on how to protect against them. It is dated 2 October 2001. The original SANS Top Ten is at http://www.sans.org/topten.htm I have inserted that list here below. I will now address each of these Top Ten vulnerabilities and what I have done to mitigate them.

Login services-- telnet (23/tcp), SSH (22/tcp), FTP (21/tcp), NetBIOS (139/tcp), rlogin et al (512/tcp through 514/tcp)

These services are denied incoming by both my router and the Sidewinder firewall. Outgoing ftp is only permitted by GIAC users through Sidewinder's ftp proxy. Outgoing telnet is permitted only by GIAC administrators.

RPC and NFS-- Portmap/rpcbind (111/tcp and 111/udp), NFS (2049/tcp and 2049/udp), lockd (4045/tcp and 4045/udp)

These services are denied incoming by both my router and the Sidewinder firewall. There are no Unix boxes in GIAC's network, so they are not needed anyway.

NetBIOS in Windows NT -- 135 (tcp and udp), 137 (udp), 138 (udp), 139 (tcp). Windows 2000 – earlier ports plus 445(tcp and udp)

These ports are open inside the GIAC corporate subnet to allow Windows inter-process communications but they are denied everywhere else.

X Windows -- 6000/tcp through 6255/tcp

X Windows is used with Unix clients to "pipe" a display from one box to another. They give the user a nice GUI interface, but it is not too secure. These services are denied incoming by both my router and the Sidewinder firewall. There are no Unix boxes in GIAC's network, so they are not needed anyway.

Naming services-- DNS (53/udp) to all machines which are not DNS servers, DNS zone transfers (53/tcp) except from external secondaries, LDAP (389/tcp and 389/udp)

DNS is permitted through the border router, but limited to Sidewinder's split DNS servers and the ISP's DNS name servers. This traffic is denied everywhere else.

Mail-- SMTP (25/tcp) to all machines, which are not external mail relays, POP (109/tcp and 110/tcp), IMAP (143/tcp)

SMTP traffic is permitted through the border router, but limited to Sidewinder's mail relays. POP and IMAP are not used and they are denied.

Web-- HTTP (80/tcp) and SSL (443/tcp) except to external Web servers, may also want to block common high-order HTTP port choices (8000/tcp, 8080/tcp, 8888/tcp, etc.)

Web traffic is permitted but closely regulated by the router's CBAC firewall features and Sidewinder's web proxies. See Section II for details.

"Small Services"-- ports below 20/tcp and 20/udp, time (37/tcp and 37/udp)

Small services are turned off at the firewall, See Section II, Border router.

Miscellaneous-- TFTP (69/udp), finger (79/tcp), NNTP (119/tcp), NTP (123/tcp), LPD (515/tcp), syslog (514/udp), SNMP (161/tcp and 161/udp, 162/tcp and 162/udp), BGP (179/tcp), SOCKS (1080/tcp)

All these services are denied incoming by both the router's CBAC filtering and the Sidewinder firewall, except syslog. Syslog (514) is permitted by proxy from the Internet, VPN and DMZ burbs to the internal subnet for monitoring purposes.

ICMP-- block incoming echo request (ping and Windows traceroute), block outgoing echo replies, time exceeded, and destination unreachable messages except "packet too big" messages (type 3, code 4). (This item assumes that you are willing to forego the legitimate uses of ICMP echo request in order to block some known malicious uses.)

I made the choice in GIAC's security policy to deny all ICMP traffic. However it may be turned on temporarily for troubleshooting purposes if there is an express need for it, by modifying the netfilter access list.

I have explained why the traffic listed above should be denied. The audit with NMAP will prove whether or not I have been successful. Although NMAP can be configured to run individual targeted checks against each of the vulnerabilities, I will take the shotgun approach and scan as many things as I can at once. Refer to Figure 9.

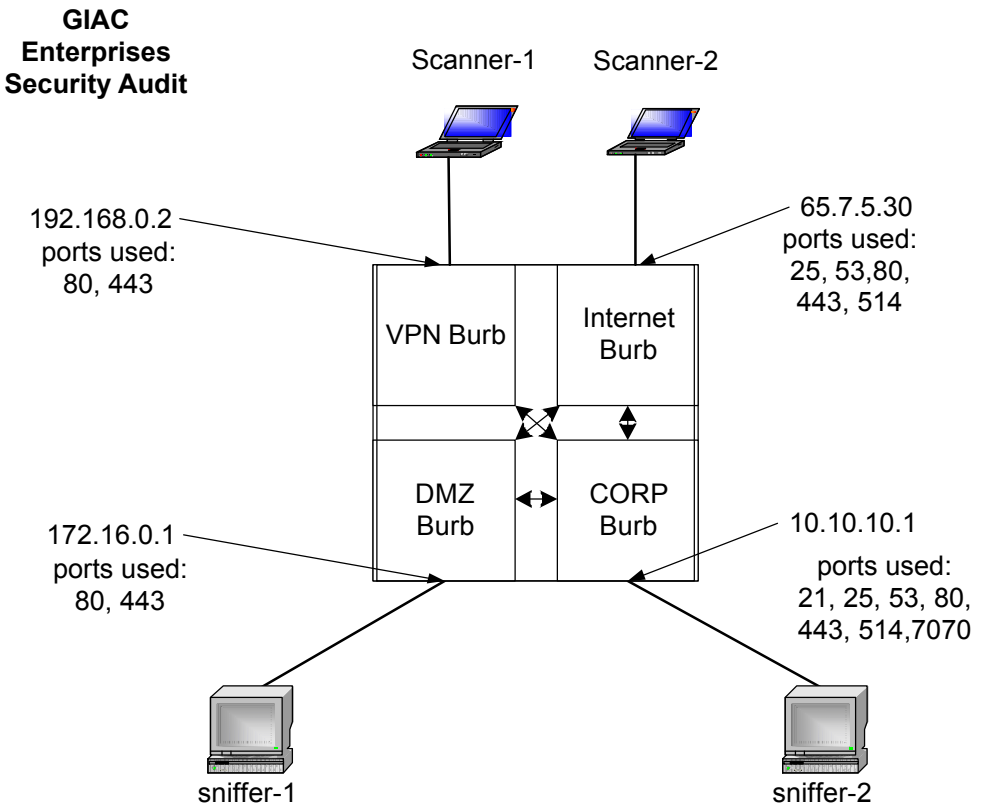


Figure 9

I would set-up “Sniffer-1” and “Sniffer-2” with Windump, which is a port of the popular tool Unix/Linux tool, tcpdump. These two boxes will be directly connected to the firewall as shown in the drawing. Tcpdump would also be running on the Sidewinder, listening on all four burbs so I could watch the action and reaction of the port scans on each firewall interface.
I will start both laptops; Scanner-1 and Scanner-2, with the following nmap port scan:
Note nmapnt is available for Windows from eEye Security
http://www.eeye.com/html/Research/Tools/nmapNT.html

```
nmap-sS -vv -p 1-2233 192.168.0.2 -oN c:\nmap\iburb-sS.txt
```

This is a stealth scan on all TCP ports up to 2233 against the VPN burb. This command uses the "-vv” switches to give me more verbose output. The "-oN" switch saves the scan log to a text file named "iburb-sS.txt" meaning Internet burb with the -sS switches set.

The expected result of these scans would be something like this:
The SYN scan took 320 seconds to scan 1523 ports.
Interesting ports on (192.168.0.2):
(The 2233 ports scanned but not shown below are in state: filtered)
Port State Service
80/tcp open http
443/tcp open https
Nmap run completed -- 1 IP address (1 host up) scanned in 321 seconds
Ports 80 and 443 are open so the Suppliers coming in from the VPN can initiate web requests to the web server.

The next port scan would be:

```
nmap-sU -vv -p 1-2233 192.168.0.2 -oN c:\nmap\iburb-sU.txt
```

This is an udp scan on all ports up to 2233 against the VPN burb**.** The naming convention continues. The expected result of these scans would be something like this:

```
Host  (192.168.0.2) appears to be up ... good.
Initiating FIN,NULL, UDP, or Xmas stealth scan against  (192.168.0.2)
The UDP or stealth FIN/NULL/XMAS scan took 290 seconds to scan 2233
ports. (no udp responses received -- assuming all ports filtered)
All 2233 scanned ports on  (192.168.0.2) are: filtered
Nmap run completed -- 1 IP address (1 host up) scanned in 295 seconds
```

The expected results for the Internet Burb scan would have http (80) and https (443) along with: Port 25, smtp for incoming mail. Port 514, syslog, and port 53 udp. Syslog data from the border router is permitted to reach the syslog station on the internal corporate burb. DNS inquiries come in on udp port 53.

While the scans were going on I would monitor the firewall’s tcpdump screens and capture the output of Windump on sniffer boxes. After the scans were complete I would

compare my Windump logs to the firewall logs. Sidewinder uses standard Unix logs such as syslog in addition to some custom reports and alerts. The logs should show traffic being denied. After the "-sS and –sU" scans were complete I would repeat the process from the inside, meaning Scanner-1 would switch places with Sniffer-1. Scanner-2 and Sniffer-2 would also switch places. Then the same two scans would be repeated.
The nmap results for the port scan against the **internal corporate burb** should be as follows: ports 21, 23, 25, 80, 443, 514, and 7070 open for ftp, telnet, http https, syslog and realmedia outgoing sessions to be requested from the internal burb. Sidewinder's proxies take care of return traffic back in. Note, only admins have outgoing telnet access.

The nmap results for the port scan against the DMZ burb should show no open ports. All http and https sessions will be requested from the Internet burb. Sidewinder's proxy will handle the return traffic back to the DMZ burb.

Analyzing the Audit

After running the NMAP scans I would have an idea what kind of shape I was in. At that point I could re-configure the acls on Sidewinder to correct any problems I may have found. I suspect during the audit, I would find some mistakes or errors of omission. They would be corrected before the firewall was installed. Nmap does not lie. If it showed open ports then I would close then.

When all this testing was complete and I felt comfortable with the results, I would then arrange to hook up the firewall on the network. I would schedule this with management in advance after normal business hours, to minimize network disruption and minimize the traffic I would be analyzing. Thursday evening would be perfect. I would also arrange with management to have a sample email account, VPN account and web access password/user name so I could test the functionality of mail, web and VPN access. At this point in time I am focusing on usability, making sure that all user services are available and working properly. I will surf the web, view Real Audio newscasts, send and receive mail from an outside email address to a local email address. I would also use the VPN test account to verify access. I would send email to and from GIAC's network. I would make sure I could actually reach all the web servers. I would test ftp and outgoing telnet capability. I would repeat the nmap scans against all the border router's ports. That way I could fine tune the router's acls and make sure it's syslog data was reaching the syslog station on the internal burb. I would make sure the web servers were talking to the databases through the proxies. I would carefully analyze all the logs to ensure the firewall was doing everything I expected it to, according to the GIAC Security policy.

Conclusion

Friday morning, when everyone came back to work, I would be on the premises to insure everything worked smoothly. Or I would fix what did not work correctly. I would have the weekend to do further work and testing as needed. I would finish the audit with a full vulnerability scan and report using ISS from my office to the GIAC network using DoS

brute force attacks over the weekend. All my scans would be saved and on file for future use. I would ask GIAC to hire me on a contract to provide monthly vulnerability testing and analysis.

His design uses Checkpoint's Firewall-1 V. 4.0. In looking at his rule set and Firewall tutorial, I see no mention of shutting down ports 256, 257, and 258 which are open by default and are used for administration. (Auditing Your Firewall Setup, Lance Spitzner ,Last Modified: 26 March, 2000)

Therefore an attack can be launched from the Internet against the firewall using specially crafted packets against those ports. For details on how this was done I turned to the work done by Dug Song, Thomas Lopatic, and John McDonald. They made an appearance at *Black Hat Briefings 2000*, and presented "an analysis of CheckPoint FireWall-1 vulnerabilities resulting from protocol design flaws, problems in stateful inspection, common or default misconfigurations, and minor implementation errors".

Their work was confirmed in their test lab and verified by Check Point, which subsequently issued patches that mitigate the vulnerabilities. See

http://www.checkpoint.com/techsupport/alerts

for service packs VPN-1/FireWall-1 4.0 SP7 and

VPN-1/FireWall-1 4.1 SP2 that eliminate each of these vulnerabilities. For

VPN-1 Appliances (IPSO) running version 4.0, the service pack is version 4.0

SP5 Hotfix.

They found the following vulnerabilities:

1. One-way Connection Enforcement Bypass
2. Improper stderr Handling for RSH/REXEC
3. FTP Connection Enforcement Bypass
4. Retransmission of Encapsulated Packets
5. FWA1 Authentication Mechanism Hole
6. OPSEC Authentication Spoof
7. S/Key Password Authentication Brute Force Vulnerability

8. GetKey Buffer Overflow

These vulnerabilities are explained briefly by ISS at this link, http://xforce.iss.net/alerts/advise62.php or in more detail by Dug Song, Thomas Lopatic, and John McDonald themselves at this web site, http://www.dataprotect.com/bh2000/ which includes the source code for the tools they created to run the attacks which they uncovered.

If I were to launch an attack against his firewall I would proceed by installing the attack code in a Red Hat 7.1 workstation. I would begin my Reconnaissance Phase

With a NMAP stealth scan to find ips that I could use to target the attack. I'd then start with the number 1 the One-way Connection Enforcement Bypass and if it failed I try the other seven attacks in order until I found one that worked.

Check Point firewall solutions use a distributed client-server model. A *management server* is used to administer the firewall and push the security policy to the firewall modules, which implement and enforce the security policy to protect the enterprise. I would focus my attacks on the communication channels between the management and firewall modules. If I can access the management module I can turn off all network security by uploading an "allow all" security policy to the firewall modules. Once that is done I can wreak havoc on the entire network at will. Un-patched Check Point firewalls are in fact susceptible to several forms of attack against these communication channels.

When the firewall is set up, the administrator defines a shared authentication secret, which is saved to disk. When an administrator wants to access the management module, the secret is exchanged transparently and access is permitted. The firewall does not check the source address of an ip attempting to authenticate, it simply refuses the connection if it does not have a secret key to match the source address. So providing I could get past the border router, I would then learn all the ip addresses for which management access has been defined. Then I could pretend to be a management module by sending spoofed packets with the ip range I gathered previously with by NMAP recon scan. That might get me management access. Another method I could try is by compromising FWN1 authentication. It works as follows according to Song, Lopatic, and McDonald:

"1. The filter module generates a random number R1.

2. The filter module signs R1. The signature S1 = Hash(R1 + K), where

'+' denotes concatenation.

3. R1 and S1 are sent to the management module.

4. The management module verifies S1.

5. The management module generates a random number R2.

6. The management module calculates the corresponding signature S2 = Hash(R2 + K), where '+' is concatenation again.

7. R2 and S2 are sent to the filter module.

8. The filter module verifies S2.

This protocol is trivially defeated by a simple replay attack. Instead of generating our own R2, we can just reuse the value R1 sent by the filter module. Then, we can also reuse the signature S1, instead of having to generate our own S2. "

I could then run their ``fw1fwn" utility to implement the unload command with FWN1 authentication.

As you can see, although I am certainly no hacker, with a little research and a lot of time I could conceivably hack into the management module, upload my own "allow all" policy and attack whomever I wanted. Such is the way *script kiddies* operate, using code and instructions from more knowledgeable crackers. These attacks are a bit more involved than the usual script kiddie attacks, but the concept is the same, i.e. a person with lots of time and attitude can cause some damage. Most admins have very little time; their duties and workloads are huge. Thus an admin may inadvertently leave himself open to attack because of an oversight or by leaving Check Point in default settings.(ports 256-258 open to the world). After disabling the firewall, I would go after the email server next. Log into it as admin and send an email virus to all users. I could use a variant to the I Love You Virus, with a different subject line such as "Attention All Hands Meeting". Coming from the administrator, the mail would certainly be read. The email code could be crafted to search and destroy critical files, or send a false emails to partners and suppliers saying they were no longer wanted. I could attack the Oracle SQL server and attempt to compromise it to gain access to the internal GIAC internal sayings database. Or I could continue my attacks described previously to attack the secondary firewall and it's trusted side to go after credit

card numbers.

Denial of Service Attack using fifty cable-modem equipped PCs

The fifty systems could be compromised by running nmap scans against @Home ip ranges to find open network shares.There are thousands of naive home users running Windows 9x, with little or no security. Back Orifice could be used to insert Wintrino, a distributed denial of service client. See link. The file name Wintrinoo is called service.exe. It is 23,145 bytes in length. When run, the program installs a copy of itself in the \windows\system directory and creates a registry entry in *HKEY_LOCAL_MACHINE/SOFTWARE/Microsoft/Windows/CurrentVersion/Run* so it will restart each time the computer is booted. The Wintrino-infected client PCs would be then become "zombies" controlled and manipulated unknowingly to the individual owner by me running Trin00 on my Linux workstation. Trin00 uses a three-tier archecture, to make it harder for the White Hat community to trace the orgin of an attack. See figure 1 below. It sends it's commands to zombie servers (Wintrinoo Masters) on port 27665 with a telnet or Netcat session. The Masters in turn send commands to the Wintrin00 client zombies on UDP port 35555. The Wintrinoo zombies will then be orchestrated to simultaneously flood GIAC's network by bombarding it with large UPD packets. The combined bandwidth load of 50 high speed PC's sending large packets to the GIAC network would seriously slow down it's traffic flow. If the Check Point firewall was running SMTP Security Server, the rapid stream of invalid SMTP commands to it would raise the CPU load on the firewall, disabling mail delivery (although other traffic continued to pass).

Ddos counter-measures

RFC 2827 (BCP-38*) Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing* is a must reading for understanding more about the mechanics of how denial of service attacks work and how to stop them. Cisco also has a document that presents much useful data at http://www.cisco.com/warp/public/707/newsflash.html (*Strategies to Protect Against Distributed Denial of Service (DDoS) Attacks,February 17, 2000).*

Although one can shut down the ports that Ddos agents commonly use, (TCP/UDP 0-20) that is not enough. Attackers can use virtually any port they wish. The best strategy entails rate-limiting packets. This should be done at both the border router and primary firewall. See Section 2 of this document for details on the countermeasures I employed with Cisco's stateful packet filtering to shut down half-open connections caused by SYN-Floods.

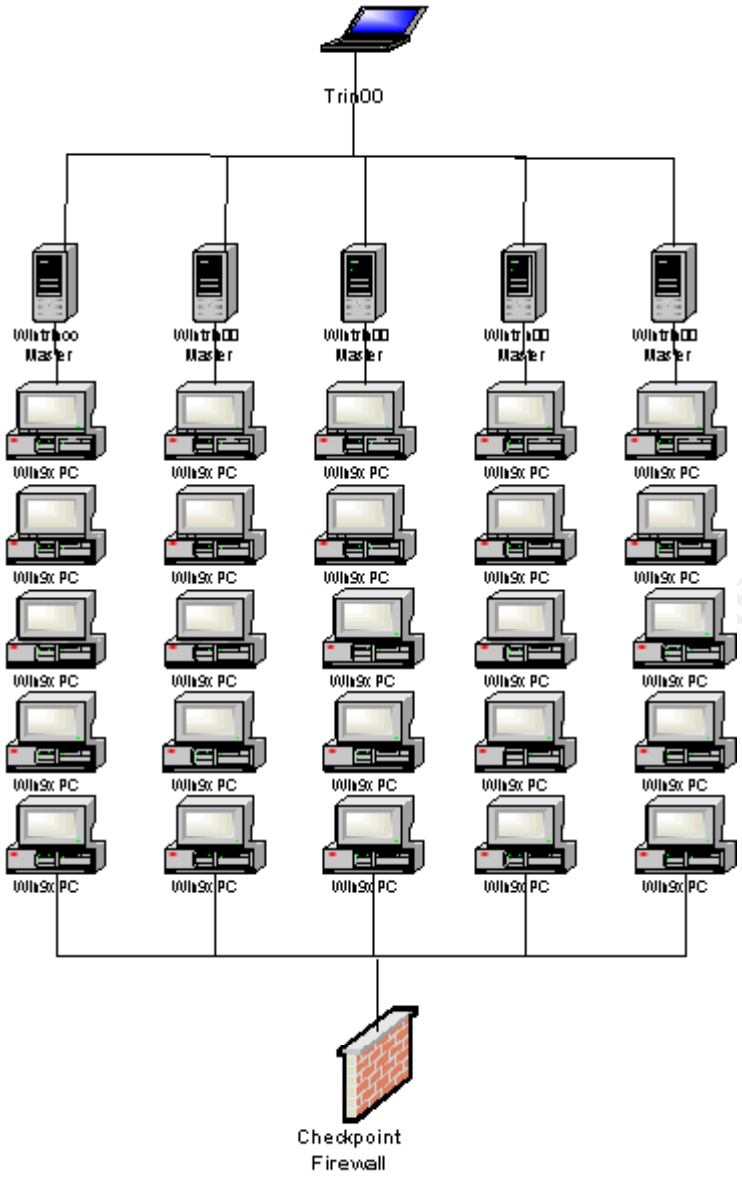


Figure 10 Distributed denial of service attack

References

Hfnetchk tool
Retrieved October 29, 2001, from
http://support.microsoft.com/support/kb/articles/q303/2/15.asp?id=303215&sd=tech

URLScan Security Tool
Retrieved October 29, 2001, from
http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/tools/URLscan.asp

IIS Lockdown Wizard
Retrieved October 29, 2001, from
http://www.microsoft.com/technet/security/tools/locktool.asp

Windows 2000 Professional Baseline Security Checklist
Retrieved October 29, 2001, from
http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/tools/w2kprocl.asp
Microsoft Personal Security Advisor
Retrieved October 29, 2001, from
http://www.microsoft.com/technet/mpsa/start.asp

Network Ice ICEpac Security Suite
Retrieved October 29, 2001, from
http://www.networkice.com/products/icepac_suite.html

SANS/FBI Top Twenty
Retrieved October 29, 2001, from
http://www.sans.org/top20.htm

Internet Security Scanner (ISS)
Retrieved October 29, 2001, from
http://www.iss.net/

Nessus
Retrieved October 29, 2001, from
http://www.nessus.org/

RFC 1918
Retrieved October 29, 2001, from
http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc1918.html

Check Point vulnerabilities

Retrieved October 29, 2001, from
http://xforce.iss.net/alerts/advise62.php

Auditing your Firewall Setup. By Lance Spitzer
Retrieved October 29, 2001, from
http://www.enteract.com/~lspitz/audit.html

ISS X-Force Alerts
Retrieved October 29, 2001, from
http://xforce.iss.net/alerts/advise62.php

Dug Song, Thomas Lopatic, and John Mconald
Retrieved October 29, 2001, from
http://www.dataprotect.com/bh2000/

Secure Computing
Retrieved October 29, 2001, from
http://www.securecomputing.com/

Compaq Proliant DL380 web page
Retrieved October 29, 2001, from
http://www.compaq.com/products/servers/proliantdl380/description.html

Intel Netstructure 3130 VPN Gateway
Retrieved October 29, 2001, from
http://www.intel.com/network/idc/products/vpn_gateway3130_tech.htm

Isolation Systems
Retrieved October 29, 2001, from
http://www.isolation.com/

Shiva Technology
Retrieved October 29, 2001, from
http://www.intel.com/network/shiva/

Password Policy Enforcer
Retrieved October 29, 2001, from
http://www.tpis.com.au/products/ppe/

Table 17 found on Cisco CBAC document
Retrieved October 29, 2001, from
http://www.cisco.com/univercd/cc/td/doc/product/software/ios121/121cgcr/secur_c/scprt3/scdcbac.htm

NMAP
Retrieved October 29, 2001, from
http://www.insecure.org/nmap/

NMAPNT
Retrieved October 29, 2001, from
http://www.eeye.com/html/Research/Tools/nmapNT.html

Check Point Alerts
Retrieved October 29, 2001, from
http://www.checkpoint.com/techsupport/alerts

Wintrino
Retrieved October 29, 2001, from
http://packetstormsecurity.org/distributed/razor.wintrinoo.txt

Strategies to Protect Against Distributed Denial of Service (DDoS) Attacks,February 17, 2000)
Retrieved October 29, 2001, from
http://www.cisco.com/warp/public/707/newsflash.html

Sidewinder Administration Guide version 5.1
March 2000
Secure Computing