Global Information Assurance Certification Paper

GIAC Enterprises: Architecture, Policy, Audit, and Under Fire
Firewalls, Perimeter Protection, and VPNs
GCFW Practical Assignment: Version 1.5d

Gale Slentz

Forward

1. Structure

This practical contains four parts. Each will be titled as:

- Assignment 1 - Security Architecture
- Assignment 2 - Security Policy
- Assignment 3 - Audit Your Security Architecture
- Assignment 4 - Design Under Fire

The first three parts are interrelated and describe the same design from various perspectives. The last part applies to a previously posted GCFW practical and uses an unrelated architecture/design.

2. Notation

To both enhance clarity and to "sanitize" information, networks and hosts will be labeled mnemonically. For example, w.x.service.0 is used to specify a service network. Its main intent is clarity of expression in a generally easy to understand format. It should not be interpreted so rigorously as to imply class B networking only. It is presented as strictly a notational convenience. A particular host on this networks is specified as w.x.serivce.web.

3. Supported Platforms

For the purposes of this assignment, GIAC Enterprises is a UNIX house. Sun Solaris and Linux are the installed platform operating systems.

1. Assignment 1 - Security Architecture

1.1 Overview of GIAC Enterprises

GIAC Enterprises and its business model can be concisely summarized by the following:

- GIAC Enterprises is a small ($200 million sales/yr.) growing Internet startup which sells online fortune cookie sayings.
- It has as just completed a merger/acquisition, and therefore has business partners.
- Acquires its sayings from a various out sourced authors.

From the knowledge of the business plan and its operations, the following types of accesses are seen to be required.

- Customers: the cookie companies that purchase the sayings.
- Suppliers: the authors of the sayings
- Partners: which include resellers and translation services.

The overall business needs of GIAC Enterprises form the architecture's functional and security requirements.

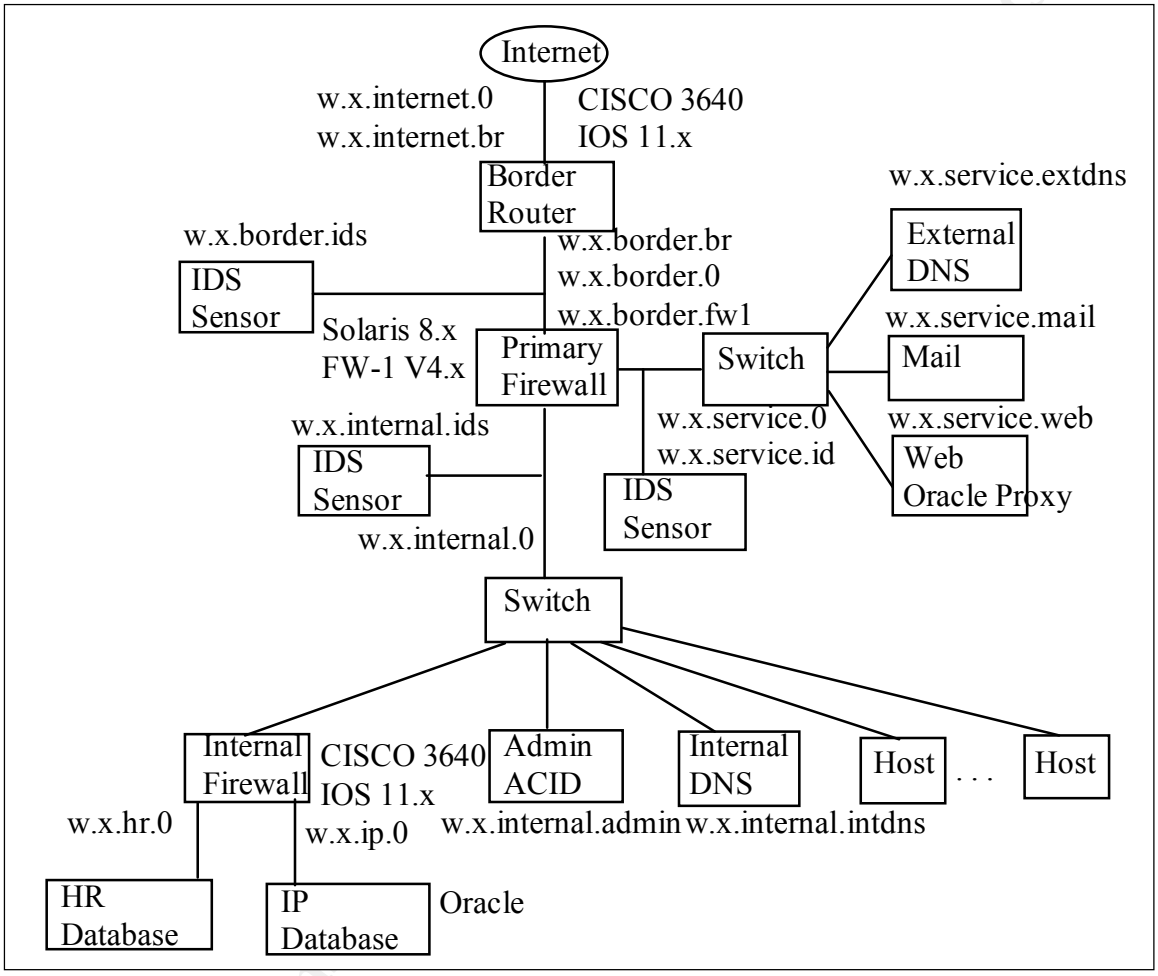
To preserve its ability to function as a business, GIAC Enterprises must protect its intellectual property and internal computing assets. The following security architecture is designed to provide the necessary perimeter defense.

1.2 Security Architecture Considerations

The overriding fundamental consideration for the security architecture, is that it "fit" the current GIAC Enterprises business practices model, providing significant security without hindering business operation. At the more specific level, any design must satisfy the following basic needs:

- Provide the required access: customers, suppliers, and partners.
- Practical: support system backups, and maintenance.
- Allow for growth: facilitate phased implementation, respond to budget realities.
- Defense in depth: Utilize the complementary natures of firewalls, network Intrusions Detection Systems (IDSs), and host IDSs.

1.3 Architecture Diagram



1.4 Architecture Component Description

1.4.1 Border Router (CISCO 4000, IOS 11.x)

The border router is the interface between GIAC Enterprises intranet and the internet. Its function is to properly provide ingress an egress packet routing of network traffic. The router's security role is to help enforce /implement GIAC's security policy. This router's security benefit is two fold. First, it protects GIAC Enterprises from the internet "noise" that is easily detected and efficiently dropped. Typically this noise consists of network mapping, port scanning, operating system (OS) fingerprinting, or simple protocol based attacks. Secondly, border filtering has the additional benefit of offloading the primary firewall, increasing its throughput and performance. The border router Access Control Lists (ACLs) should filter the following:

- ✂Private IP addresses: 0.0.0.0, 127.0.0.0-127.255.255.255, 10.0.0.0 - 10.255.255.255, 172.16.0.0-172.31.255.255, 192.168.0.0-192.168.255.255.
- ✂Spoofed packets: Those inbound from the internet but containing GIAC Enterprise's network IP addresses as source addresses.
- ✂Unknown protocols: Expected protocols are ICMP, UDP, and TCP, so anything else is suspect.
- ✂Potentially misused protocols and options: Source routing, broadcasts, Land attacks.

Second, it must keep internal information inside, both to protect from inadvertent escape and to be a "good Internet neighbor".

- ✂Spoofed packets: Those exiting from CIAC's network, but containing source IP addresses not in GIAC's network space.

Note that the border router is used as a fast and efficient pre-filter. It inspects packets one by one with no concept of previously established UDP or TCP communication, and has no knowledge of protocol at the application level.

1.4.2 Primary Firewall (Check Point's FW-1 V4.x on Sun Solaris 8.x)
The primary firewall is the interface between GIAC's border network (w.x.border.0) and GIAC's service network (w.x.service.0) and internal network (w.x.internal.0). Like the border router, the primary firewall must properly route packets from/to each of the three networks that it services.

The firewall is used to isolate the service network from the internal GIAC Enterprise network. This allows for more tuned and application specific ACLs to be applied to the service network. The service network needs this added care and protection since it is the most exposed to potentially malicious and anonymous internet users. As a side benefit, this isolation of sub-nets makes writing, testing, and maintaining the router ACLs easier.

The primary firewall's security role is to use its stateful knowledge of IDP and TCP communication/connection state to block, log, and alert to security policy violations. The knowledge of connection state allow the blocking of the more sophisticated probes that include:
✂Undefined TCP flag combinations
✂Fragmented packet probes
✂ UDP unsolicited traffic

It also contains the more specific knowledge of which network and services are allowed access privilege. It is hardened to bastion host level.

1.4.3 Switch

Both the service network (w.x.service.0) and the internal network (w.x.internal.0) are switched. The internal network switch is used primarily to increase network performance on the internal network. However, both switches have a security role, which is to reduce the effectiveness of attacker installed network sniffers. Switches are to be used to mitigate the damage done by a possible compromised system. If an attacker succeeds in

rooting a system, their typical next step is to start a "sniffer" in an attempt to intercept additional account/password pairs from the network traffic. If successful, this technique often leads to a chain of successively compromised systems on that network.

1.4.4 Network Intrusion Detection System (IDS) Sensors

The IDS sensors are placed to monitor network traffic flowing in/out of both the service network (w.x.service.0) and the internal GIAC network (w.x.internal.0). This allows them to see all traffic that survives the border router and stateful firewall. Their security function is to provide payload based attack detection in conjunction with general real time network traffic knowledge. They also provide a check for misconfiguration of the border router and the stateful firewall, and will be utilized in the periodic auditing of the GIAC security perimeter. It is planned that IDS discovered events will be logged to a central station (w.x.internal.admin) allowing archiving and ad hoc querying of the event data base. The IDS sensors will normally run SNORT to provide the IDS functionality. Snort will be run in Network Intrusion Detection System (NIDS) mode. In this mode Snort operates from a set of rules, analyzing network traffic in real-time, and generating alerts. Additionally, each IDS sensor will have the tcpdump application installed to monitor network traffic during audits or ongoing intrusion investigations.

1.4.5 External DNS Server

This server's function is to provide DNS to the internet users. It contains resource records only for the three servers (external DNS, mail, web) on the service network. Split horizon DNS is used by GIAC Enterprises to reduce/eliminate internal network/host information from potential attackers. It is hardened to bastion host level.

1.4.6 Mail Server

This server provides SMTP to both internet and GIAC users. SMTP is used as the transfer agent in/out of GIAC. POP3 is available to support internal users mail download (through Netscape Communicator Messenger browser) to their individual hosts for reading. In this way, w.x.service.mail never initiates a session into the internal network. When sending, internal user's browser application will forward to the service network SMTP mail server where it gets processed and potentially forwarded to other internet mail handlers.
Its security role is to provide mail services in/out of GIAC from a bastion host hardened server.

1.4.7 Web Server

The web serve's function is to provide the GIAC Enterprise's web presence and business capability to its customers, partners, and suppliers. All GIAC's business applications are web based and interface with the internal Intellectual Property (IP) database (Oracle). An Oracle ASO (proxy like front end) runs on the web server host.

The web server is of particular security concern, since it is the main communication point for GIAC. It must be well protected. All business information should be transferred to the internal IP database as soon as possible to minimize the amount and duration of information on the web server itself.

The web server's security role is to provide VPN functionality as needed for customer, partner, and supplier GIAC interaction. This is accomplished through HTTPS/SSL (for authentication and network traffic encryption) in conjunction with an encrypting Oracle proxy (ASO) from this bastion host hardened server.

GIAC's host hardening administration expertise is in UNIX, so this host will be implemented on a Sun Solaris platform. Since buffer overrun attacks are considered a primary risk, the kernel will be built and configured to disallow execution from its stack. To secure the web server against any stack based buffer overrun attack the following two lines are added to /etc/system:

```
set noexec_user_stack = 1
set noexec_user_stack_log = 1
```

1.4.8 Internal DNS

The internal DNS server's function is to provide DNS to the GIAC Enterprise's internal users. It contains the resource records for all hosts on the GIAC intranet (w.x.GIAC.0). This information is for internal use only and must not be advertised/provided to the external internet.

Its security role is to provide the other half of the split DNS functionality to eliminate the availability of internal network/host information from potential external attackers. It is hardened to bastion host level.

1.4.9 Administration Host

The administration host has three jobs. The first is to provide a hardened host from which administration can more safely be accomplished. It is the means for accomplishing backup and configuration of the service network servers. Additionally, it serves as the IDS sensor's database and analysis station for the three Snort sensors. Analyst Console for Intrusion Databases (ACID), with a mySQL database module, will be used to store, query, and generally analyze network events. And finally, it serves as the log host for the boundary router, primary firewall, and internal firewall.

Its security role is to provide a more secure, hardened platform from which administration and intrusion analysis is done. SSH (and its scp command) is the basis for remote administration in order to provide network encrypted authentication (no open text passwords) and data file transfers.

1.4.10 Internal Firewall (CISCI 3640, IOS 11.x)

The role of the internal firewall is to provide limited and controlled access to the Human Resources (HR) database and the Intellectual Property (IP) database. Its security role is to restrict access from unauthorized use by both insiders and outsiders.

There are two separated database interfaces to isolate them from each other. This allows the maximum flexibility in specifying the ACLs differently for the two databases.

1.4.11 IP Database

The IP database is used to store the supplier cookie sayings, the customer orders, the partner business dealings, and and related GIAC business data. These business transactions are handled through GIAC's web application software, with authentication and network encryption provided by the HTTPS protocol, and database transaction traffic encryption and interface provided by the Oracle ASO package.

1.4.12 HR Database

The HR database is used to store/access any human resource function data. These typically involve payroll, bonuses, awards, etc.

1.4.13 Hosts

User hosts interact at the web level via Netscape Communicator (Navigator and Messenger). Mail is retrieved/send from/to the GIAC mail server (w.x.service.mail).

1.5 Security Architecture Supported Accessibility

As an internet startup, GIAC has established its business applications to be web centric. The underlying philosophy is to use HTTPS/SSL web protocol to provide the needed confidentiality, and authentication security functions. One major benefit is the ubiquity of the client software (Netscape browser). This minimizes or eliminates support, installation, maintenance and training costs. The Netscape browser's has built-in the required VPN capabilities. Additionally, it provides the fine grained control of various security settings.

1.5.1 Customers

It is envisioned that customers will access the site, browse and ship, enter charge card information, all through the web browser just as any other web shopper does today. The site will be secured through the use of SSL to encrypt the private information going back and forth over the internet. One sided SSL only will be used. What this means is that the customer's browser receives a copy of the Verisign signed GIAC company certificate. This can be verified and or viewed by the customer. The customer does not prove their identity per-se, but does provide a valid (and verifiable) credit card account number.

1.5.2 Suppliers and Partners

Due to the web based nature of GIAC, its business applications are accessed via the web. This places extra emphasis on the security of their web server. As a small startup, GIAC has historically assigned each supplier and partner with "strong" passwords to authenticate with the web server ("basic password authentication"). This function was accomplished by running a computer password generator, generating security policy compliant passwords which are then assigned to each supplier/partner user. Initially, this practice will continue. However, as GIAC's growth continues and the internet threat escalates, it is advisable to investigate the use of one time password generating tokens (such as SecureID). These could be issued to suppliers and partners. See Assignment 3 - Audit Your Security Architecture for more information.

2. Assignment 2 - Security Policy

2.1 Overview of Security Policy

At the topmost level, a site's security policy is an attempt to describe what computing resources are available to whom, and by what acceptable means. Additionally, the rights and responsibilities of the various users and administrators are described. Typically, a default deny based policy is the more secure and implementation bias. Individual services are then permitted based on business need and risk assessment.

An interpretation of this policy is what gets implemented by the various elements forming the network security components. Different elements are used individually or in conjunction with other to satisfy each of the particular constraints.

2.2 Security Policy Considerations

The idea is to apply the combination of risk and value measures to generate the policy components. Critical servers on the DMZ are both high valued and high risk. Internal HR and IP databases are of very high value. Areas of high risk are unpatched user hosts, clear text passwords, and inter-host trust relationships.

2.3 GIAC Enterprise's Security Policy

- Default deny bias. (Implemented by configuring the border router, primary firewall, and internal firewall only allow what is specifically required, all others are denied)
- All connections from/to the internet must go through the GIAC firewall.
- Internal network addressing and structure should be hidden from the internet. (implemented with split horizon DNS)
- Critical server integrity and availability is required. (implemented by bastion host level configuration on service network, and biannual security audits of the security perimeter and configuration of critical servers.)
- No clear text passwords are to go over the GIAC network or out on the internet. (Implemented through administration use of SSH and its family of commands. HTTPS/SSL is used for business applications)
- Apply due diligence of internet use. (implemented by installing IDS and employing "good internet neighbor" anti-spoofing packet filtering at the border router.)
- All e-mail should go through GIAC's mail server. (implemented by blocking at the primary firewall.)

2.4 Border Router Security Policy Specifics

It is the border router's responsibility to help enforce the GIAC policy of default deny, and due diligence of internet use.

- Block entering or exiting traffic of the obvious "noise" variety:
  - Deny entering traffic with GIAC's source IP address space. This indicates spoofing or a possible unauthorized network "leak" to the internet.
  - Deny exiting traffic with non_GIAC source IP addresses. This is the "good internet neighbor" rule, so that your hosts don't participate in spoofing or distributed DoS attacks.
  - Deny ICMP from exiting the GIAC intranet. This eliminates information back to potential scanners.
  - Deny unroutable and illogical IP addresses from entering or exiting GIAC. This could indicate attempted scans from the outside.
  - Deny traceroute and other TCP options from entering GIAC. This eliminates another way of attackers gaining network and host information.
  - Deny unknown protocols. This implements the default deny bias.
- Permit GIAC entering traffic, of the allowed service, to the allowed server on the service network only.
- Permit GIAC exiting traffic, of the allowed service, to the allowed destination, from the service network only.

Logging is good, but only up to a point. There are really two consideration here. Excess logging slows the throughput of the router, so from a performance standpoint, logging may need to be reduced. Sometimes too much logging can obscure the important events, as they get lost or unnoticed in all the noise. On the other hand, good logs help in test, audit, and event investigation. This ends up being a tradeoff and requires judgment and experience to get it right.

2.4.1 Border Router ACL Syntax for Extended Rules

access-list <a number from 100-199> permit | deny protocol (one of sourceIP,host,any) [sourceMask] destinationIP [destMask] [(one of range, eq, gt)] {log]

2.4.2 Border Router Rules Ordering Tutorial

Each rule in an access list is checked from top down, and at the first encountered match, the specified action (permit/deny) is applied to the packet. Then the next packet is checked restarting at the top of the access list. If no matches are found in the access list, there is a default deny applied.

2.4.3 Border router Tips and Gotchas

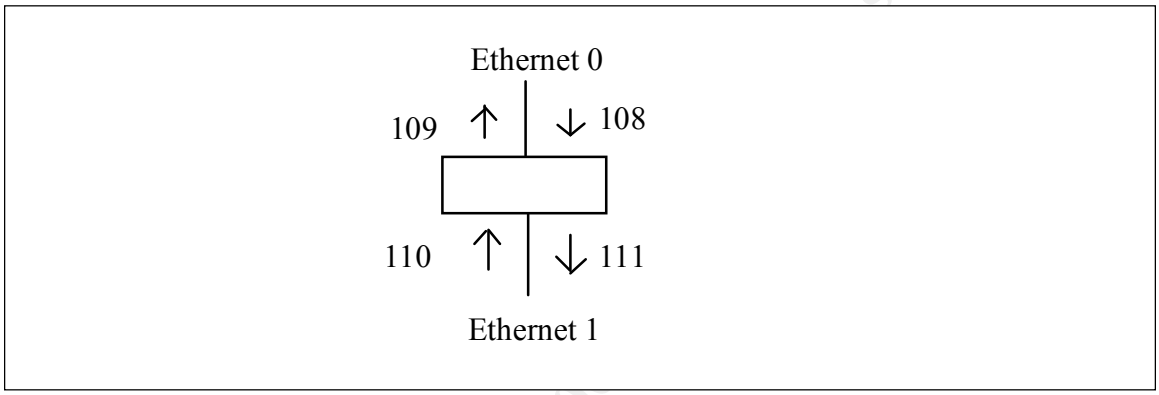
With respect to rule ordering, since ACLs are checked from top to bottom, with a default deny at the end, place more specific rules at the top. Balance the need for understandable/maintainable access lists with the need for performance efficiency. High

match rate rules should be near the top for increased efficiency.

Balance logging because of the trade off between throughput, event tracking and obfuscation.

Often filtering can be done at either the router's input or its output interface. So eliminate unnecessary CPU cycles by applying the filter at the input, so packets don't go through the packet forwarding process only to get filtered out.

2.4.4 Border Router ACLs




```
interface Ethernet 0
        ip address w.x.internet.br 255.255.255.0
        ip access-group 108 in
        ip access-group 109 out
interface Ethernet 1
        ip address w.x.border.br 255.255.255.0
        ip access-group 110 in
        ip access-group 111 out

Inbound Internet Traffic
! Inbound from the border router's internet network interface (w.x.internet.br)
! deny inbound ip with GIAC source IP, spoofed packet.
access-list 108 deny ip w.x.GIAC.0 0.0.0.255 log
! deny non-routable source IPs.
access-list 108 deny ip 127.0.0.0  0.255.255.255 log
access-list 108 deny ip host 0.0.0.0 log
access-list 108 deny ip 10.0.0.0  0.255.255.255 log
access-list 108 deny ip 172.16.0.0  0.15.255.255 log
access-list 108 deny ip 192.168.0.0  0.0.255.255 log
access-list 108 deny ip 224.0.0.0  15.255.255.255 log
! permit ICMP pings from partners to web server.
access-list 108 permit icmp w.x.partner.0  0.0.0.255 host w.x.service.web echo-request
log
```

```
! permit ICMP ping responses back from any.
access-list 108 permit icmp any w.x.GIAC.0  0.0.0.255 echo-reply log
! permit http to web server from any.
access-list 108 permit tcp any host w.x.service.web equ 80 log
! permit https to web server from any.
access-list 108 permit tcp any host w.x.service.web eq 443 log
! permit DNS to external DNS from any.
access-list 108 permit udp any host w.x.service.extdns eq 53 log
! permit mail to mail server from any.
access-list 108 permit tcp any host w.x.service.mail eq 25 log
! deny remaining ip and log
access-list 108 deny ip any any log
! default deny all remaining

! have done all filtering at the inbound Internet interface, defined by access-list 108
access-list 111 permit ip any any
!default deny all remaining

Outbound Internet Traffic
! deny non-routable source IPs.
access-list 110 deny ip host 0.0.0.0 any log
access-list 110 deny ip 10.0.0.0  0.255.255.255 any log
access-list 110 deny ip 172.16.0.0  0.15.255.255 any log
access-list 110 deny ip 192.168.0.0  0.0.255.255 any log
access-list 110 deny ip 224.0.0.0  15.255.255.255 any log
! permit GIAC hosts to ping out to anywhere
access-list 110 permit icmp w.x.GIAC.0  0.0.0.255 any echo-request log
! permit GIAC web server to respond out to ping request from partners
access-list 110 permit icmp w.x.service.web w.x.partner.0  0.0.0.255 echo-reply log
! permit web responses to anywhere
access-list 110 permit tcp host w.x.service.web any established log
! permit external dns server to respond out to anywhere
access-list 110 permit udp host w.x.service.extdns any log
! permit mail to transfer out to anywhere
access-list 110 permit tcp host w.x.service.mail any established log
! deny remaining ip and log
access-list 110 deny ip any any log
! default deny all remaining

! have done most filtering at the inbound Internet interface, defined by access-list 110
! must permit desired echo-requests and echo-responses
access-list 109 permit icmp host w.x.service.web w.x.partner.0 0.0.0.255 echo-reply log
access-list 109 permit icmp w.x.GIAC.0  0.0.0.255 any
! keep this router from responding to any icmp itself when IT is scanned!!
! and don't respond with any icmp com. admin. prohibited by filtering messages.
access-list 109 deny icmp any any
```

access-list 109 permit ip any any established
!default deny all remaining

2.4.5 Border Router Hardening

To harden this border router and satisfy the last of the security policy items that apply, the following commands should be issued:

CISCO command	*Purpose*
no ip direct-broadcast	This denies directed broadcasts, so smurf attacks and easy network/host mapping techniques are blocked.
no ip source-route	This blocks address spoofing techniques that try to force responses back through the spoofer host (very useful for session highjacking attacks).
no service finger	Standard elimination of unnecessary services.
no service tcp-small-servers	Standard elimination of unnecessary services.
no service udp-small-servers	Standard elimination of unnecessary services.
logging w.x.internal.admin	Since the CICSO router has no disk, off host syslogging is used.
no snmp	Block all SNMP traffic

2.4.6 Border Router ACL Testing

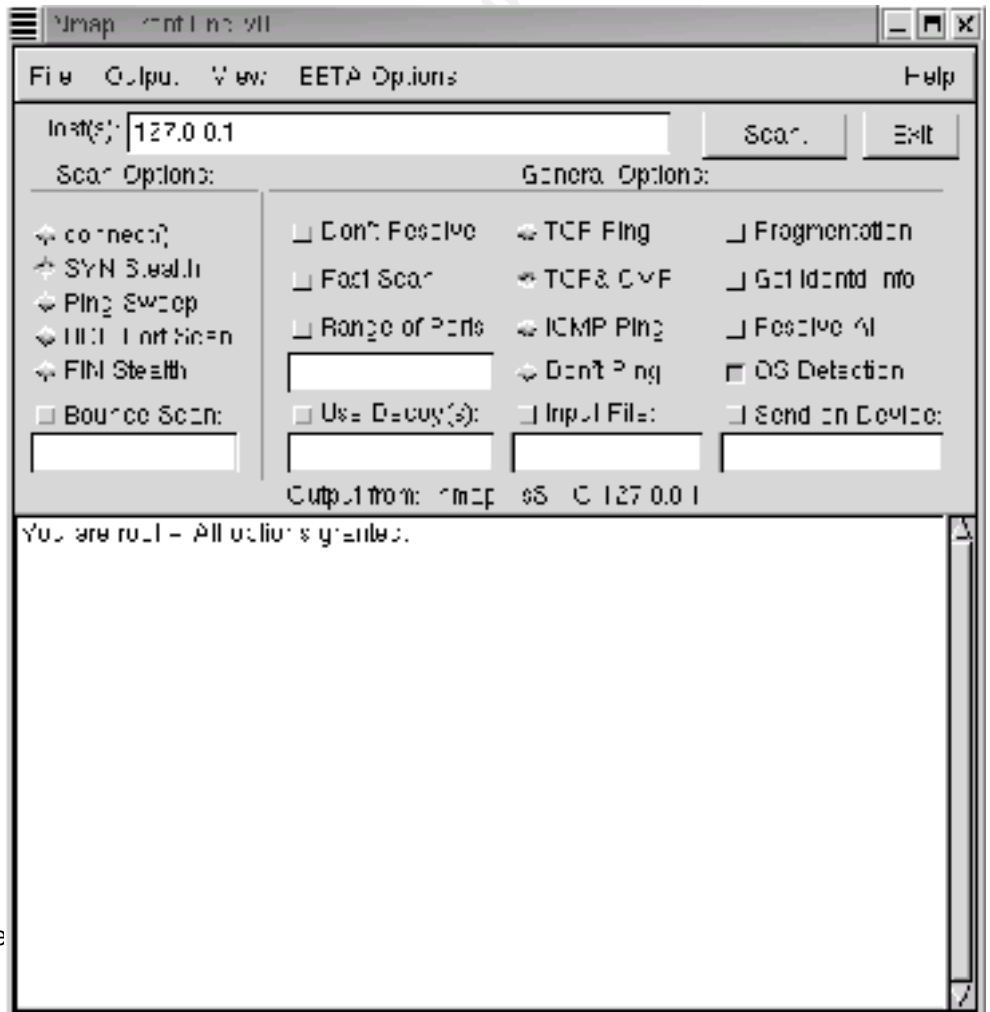
The nmap tool will be used to send the stimulus packets through the border router to the various CIAC servers. Xnmap is nmap with a GUI front end and is pictured immediately below:

xnmap

The border firewall element can be tested individually by running nmap from a host on one side of the border router's network interfaces and then checking the router's log messages for the expected permit/deny. For the paranoid, or to unravel a particularly knotty problem, tcpdump can also be used to examine the network traffic itself. The IDS sensors are handy places to run tcpdump from. To invoke tcpdump and have it snif all packets:

```
tcpdump -lvnx -s 128
```

This command produces a stdout line buffered, verbose (without address conversion) hex output of the first 128 bytes.



To test the w.x.service.web oriented filters:
From a partner's host w.x.partner.host, target w.x.service.web.

Test explanation	*nmap command*	*Expected nmap result*	*Expected router log*	*tcpdump at w.x.border.ids*
Run full udp port scan	Nmap -sU -p 1-65535 -P0 w.x.service.web	No port information on w.x.service.web (since no icmp com. admin. denied from router)	list 108 denied udp w.x.partner.host(x)->w.x.service.web(all ports)	No packets
Run full tcp SYN port scan	Nmap -sS -p 1-65525 -P0 w.x.service.web	80/tcp open http 443/tcp open https	list 108 permitted tcp w.x.partner.host(x)->w.x.service.web(80) list 108 permitted tcp w.x.partner.host(x)->w.x.service.web(443) list 108 denied tcp w.x.partner.host(x)->w.x.service.web(all other ports)	TCP SYN packets to 80,443 TCP SYN/ACK returned for 80, 443.
Run icmp ping with a decoy of an unassigned internet IP	Nmap -sP -P0 -D w.x.unassigned.host w.x.service.web	Host w.s.service.web appears to be up	List 108 permitted icmp w.x.partner.host->w.x.service.web list 108 denied icmp w.x.unassigned.host->w.x.service.web list 109 permit icmp w.x.service.web->w.x.parter.host list 110 permit icmp w.x.service.web->w.x.parter.host	Only ICMP ping request from w.x.partner.host only ICMP ping response to w.x.partner.host request

To test the w.x.service.extdns oriented filters:
From a partner's hostw.x.partner.host, target w.x.service.extdns.

Test explanation	*nmap command*	*Expected nmap result*	*Expected router log*	*tcpdump at w.x.border.ids*
Run full udp port scan	Nmap -sU -p 1-65535 -P0 w.x.service.extdns	53/udp open dns	list 108 permitted udp w.x.partner.host(x)->w.x.service.extdns(53) list 108 denied udp w.x.partner.host(x)->w.x.service.extdms(all other ports)	UDP packet sent to port 53
Run full tcp SYN port scan	Nmap -sS -p 1-65525 -P0 w.x.service.extdns	No port information for w.x.service.extdns (since no icmp com. admin. denied from router)	list 108 denied tcp w.x.partner.host(x)->w.x.service.extdns	No packets
Run icmp ping	Nmap -sP -P0 w.x.service.extdns	Host w.s.service.extdns appears to be down	list 108 denied icmp w.x.partner.host->w.x.service.extdns	No packets

To test the w.x.service.mail oriented filters:
From a partner's host w.x.partner.host, target w.x.service.mail.

Test explanation	*nmap command*	*Expected nmap result*	*Expected router log*	*tcpdump at w.x.border.ids*
Run full udp port scan	Nmap -sU -p 1-65535 -P0 w.x.service.mail	No port information on w.x.service.mail (since no icmp admin. com. denied by router)	list 108 denied udp w.x.partner.host(x)->w.x.service.mail(all ports)	No packets
Run full tcp SYN port scan	Nmap -sS -p 1-65525 -P0 w.x.service.mail	25/tcp open http	list 108 permitted tcp w.x.partner.host(x)->w.x.service.mail(25) list 108 denied tcp w.x.partner.host(x)->w.x.service.mail(all other ports)	TCP SYN packets sent to port 25 TCP SYN/ACK returned for port 25
Run icmp ping	Nmap -sP -P0 w.x.service.mail	Host w.s.service.mail appears to be down	list 108 denied icmp any->w.x.service.mail	No packets

To test spoofed and private (non-routing) addresses:
From a partner's host w.x.partner.host, target w.x.service.web.

Test explanation	*nmap command*	*Expected nmap result*	*Expected router log*	*tcpdump at w.x.border.ids*
Run full tcp SYN port scan with a decoy of a GIAC w.x.internal.host	Nmap -sS -p 80 -P0 -D w.x.internal.host 127.0.0.1,10.10.10.1,172.16.0.1,192.168.0.1,224.0.0.1 w.x.service.web	80/tcp open http	List 108 permitted icmp w.x.partner.host->w.x.service.web list 108 denied icmp (decoyhosts)->w.x.service.web	Only TCP SYN packet from w.x.partner.host should be seen

2.5 Primary Firewall Security Policy Specifics

It is the primary firewall's responsibility to enforce the GIAC security policy by permitting only the allowed accesses to the allowed services. The only services offered to the general internet are the web, mail, and external dns servers on the service network.

1. Provide dedicated web, mail, and external dns services from bastion hosts.
2. Provide SSH to the w.x.internal.admin host to allow for secure administration of the servers.
3. All e-mail should go through GIAC's mail server.
4. Allow http and https access to the web server
5. Allow GIAC internal hosts access to any web server
6. Allow w.x.partner.0 to ping the web server.
7. Allow the Oracle proxy on the web server to communicate with the internal Oracle IP database.
8. Allow syslog from the IDS sensors, border firewall, and primary firewall to the w.x.internal.admin host.

2.5.1 Primary Firewall Security Policy Implementation

The primary firewall will be Check Points FireWall-1. It will provide dynamic (stateful) communication session capability. Individual rules will be created to apply the default deny bias. The allowed services and authorized users will be accommodated with permit rules. FireWall-1 will be installed on a bastion host hardened Sun Solaris 2.8.x with non-executable stack settings to eliminate buffer overrun attacks.

2.5.2 Primary Firewall Rules Tutorial

Ordering is crucial, because the rules are checked from the top down, looking for the first match.

- Define the FW-1 objects from knowledge of the security plan and perimeter architecture. (see example objects below)
- Define the FW-1 rules necessary to implement each element of the security plan. Specify the rule in the RW-1 policy editor by selecting Edit->Add Rule
- Repeat 2 until full rule set is built
- The deny all rule should be the last rule for the firewall.
- Select Policy->Install to finally compile and install the ruleset in the firewall.

2.5.3 Notes and Tips

- If you need to log, log long to provide enough information to analyze.
- Have the on host logging log into a dedicated file system partition so that if it fills, the system remains up and running.
- To increase throughput, don't use name resolution in the logs.
- You may need to use the different "views" to adequately analyze an event from the logs. Active view and accounting views are available.
- To gain explicit control over logging and acceptance in our Policy Editor rules, remove the defaults by selecting Policy->Properties then uncheck
    - Accept Outgoing Packets
    - Accept RIP
    - Accept Domain Name Queries (UDP)
    - Accept Domain Name Queries (TCP)
    - Accept Domain Name Download

2.5.4 Defining the FW-1 Objects By Example

Networks: border, service, internal, hr, ip.
Services: domain-udp, pop-3, ssh, syslog, smtp, http, https, oracle
Routers: fw1
Groups: ids-group
Hosts: admin, extdns, web, mail, intdns.
Objects: Networks, Services, Routers, Firewalls.

2.5.5 Primary Firewall Rules

#	*src*	*dst*	*service*	*action*	*track*
#1	Any	fw1	Any	drop	Long
#2	fw1	admin	syslog	accept	Long
#3	admin	service	ssh	accept	Long
#4	Any	extdns	domain-udp	accept	Long
#5	internal	mail	pop-3	accept	Long
#6	Any	mail	smtp	accept	Long
#7	mail	Not internal	smtp	accept	Long
#8	Any	web	http	accept	Long
#9	Any	web	https	accept	Long
#10	internal	Any	http	accept	Long

#	*src*	*dst*	*service*	*action*	*track*
#11	internal	Any	https	accept	Long
#12	ids-group	admin	syslog	accept	Long
#13	web	ip	oracle	accept	Long
#14	web	Any	Any	reject	Alert
#15	mail	Any	Any	reject	Alert
#16	extdns	Any	Any	reject	Alert
#17	Any	Any	Any	drop	Long

2.5.6 Rules explanation

#	*explanation*
#1	The firewall lock down rule, admin. done from firewall console only.
#2	Allows the Firewall to log off-host to the internal admin host
#3	Allows the internal admin host to ssh to the servers
#4	Allow any user DNS access to extdns server
#5	Allow internal users to get mail from mail server
#6	Allow any user to send mail to mail server on smtp
#7	Allow mail server to transfer mail out to internet
#8	Allow any user access to web server on http
#9	Allow any user access to web server on https
#10	Allows internal hosts to access any http server
#11	Allows internal hosts to access any https server
#12	Allows the ids-group of ids hosts to syslog to the admin host
#13	Allows the web server Oracle proxy to access the ip data base
#14	Rejects and alerts any attempts by web server to initiate non-allowed connections, (coupled with rule 13.)
#15	Rejects and alerts any attempts by mail server to initiate internal connections, (coupled with rule 7.)
#16	Rejects and alerts any attempts by external dns to initiate any communications
#17	Drops and logs all remaining fall through (unauthorized) traffic

2.5.7 Primary Firewall Rule Testing

Since there are three subnetworks serviced by the primary firewall, we must generate stimuli from each. This is easily accomplished by running nmap from each of the IDS sensor hosts. Both the FW-1 logs and the nmap responses should be analyzed for the expected results.

From w.x.border.ids:

Rule #	*Test*	*Expected nmap Result*	*Expected FW-1 Logs*
#1	Run full tcp SYN scan of w.x.border.fw1 nmap -sS -p 1-65535 -P0	No open ports	All Rule #1 drop logs from source w.x.border.ids

Rule #	*Test*	*Expected nmap Result*	*Expected FW-1 Logs*
	w.x.border.fw1 Run full udp scan of w.x.border.fw1 nmap -sU -p 1-65535 -P0 w.x.border.fw1	No open ports	All Rule #1 drop logs from source w.x.border.ids
#2	Just by running these tests.	-	Logs arrive at the w.x.internal.admin analysis station
#4	Run udp scan of w.x.service.extdns port 53 nmap -sU -p 53 -P0 w.x.service.extdns	53/udp dns	1 Rule #4 accept log from source w.x.border.ids
	Run tcp scan of w.x.service.extdns port 53 (no zone transfers) nmap -sS -p 53 -P0 w.x.service.extdns	no open port	1 Rule #17 drop log from source w.x.border.ids
#6	Run tcp scan of w.x.service.mail port 25 nmap -sS -p 25 -P0 w.x.service.mail	25/tcp mail	1 Rule #6 accept from source w.x.border.ids
#8	Run tcp scan web port 80 nmap -sS -p 80 -P0 w.x.service.web	80/tcp http	1 Rule #8 accept from source w.x.border.ids
#9	Run tcp scan web port 443 nmap -sS -p 443 -P0 w.x.service.web	443/tcp https	1 Rule #9 accept from source w.x.border.ids
#12	By running test for Rule #1 above should get scan alarms from w.x.border.ids	-	2 Rule #12 accepts from source w.x.border.ids
#17	Run tcp scan web port 23 nmap -sS -p 23 -P0 w.x.service.web	No open ports	1 Rule #17 drop from source w.x.border.ids

From w.x.service.ids

Rule #	*Test*	*Expected nmap Result*	*Expected FW-1 Logs*
#7	Run tcp scan w.x.partner.mail port 25 nmap -sS -p25 -P0 w.x.partner.mail	25/tcp mail	1 Rule #7 accept log from source w.s.service.ids
#14	Run tcp scan w.x.partner.mail with decoys web, mail, extdns port 25		1 Rule #17 drop log from source

Rule #	*Test*	*Expected nmap Result*	*Expected FW-1 Logs*
#15 #16	nmap -sS -p25 -P0 -D w.x.service.web, w.x.service.mail,w.x.service.extdns w.x.partner.mail	25/tcp mail	w.x.service.ids 1 Rule #14 Alert log from source w.x.service.web 1 Rule #15 Alert log from source w.x.service.mail 1 Rule #16 Alert log from source w.x.service.extdns
#13	Run tcp scan w.x.ip.ip with decoy w.x.service.web port oracle nmap -sS -poracle -P0 -D w.x.service.web w.x.ip.ip	No open port	1 Rule #13 accept log from source w.x.service.web 1 Rule #17 drop log from source w.x.service.ids

From w.x.internal.ids:

Rule #	*Test*	*Expected nmap Result*	*Expected FW-1 Logs*
#10	Run tcp scan w.x.service.web port 80 nmap -sS -p80 -P0 w.x.service.web	80/tcp http	1 Rule #10 accept log from source w.x.internal.ids
#11	Run tcp scan w.x.service.web port 443 nmap -sS -p443 -P0 w.x.service.web	443/tcp https	1 Rule #11 accept log from source w.x.internal.ids
#3	Run tcp scan w.x.service.web with decoy w.x.service.admin port 22 nmap -sS -p22 -P0 -D w.x.service.admin w.x.service.web		1 Rule #3 accept log from source w.x.service.admin 1 Rule #17 drop log from source w.x.internal.ids
#5	Run tcp scan w.x.service.mail port 110 nmap -sS -p110 -P0 w.x.service.mail	110/tcp pop3	1 Rule #5 accept log from source w.x.internal.ids

The above procedure only tests the primary firewall's rules. It is not an end to end application function test. To test to that level, keep in mind the following utilities:

- mail: just apply a command like:
  echo "End to End mail test" | mail auditor@w.x.service.mail
- dns: remember the full functionality of nslookup.
- http: just telnet to establish an initial connection.
  Telnet w.x.service.web 80
- icmp: ping

2.6 VPN Security Policy Overview

Analysis of GIAC's working business model in conjunction with its business relationships with customers, partners, and suppliers supports leveraging their current e-business web application to effect VPN. All business applications which require the properties of VPN will be run under HTTPS/SSL. This underscores the critical nature of the web server along with the main Oracle database information.

2.6.1 VPN Security Policy

- All internet and service network business traffic with customers, suppliers and partners is to be strongly encrypted. (This is to be implemented by the application's use of HTTPS/SSL. Specifically SSL version 3 with either "RC4 with 128 bit encryption and MD5 message authentication" or "Triple DES with 168 bit encryption and SHA message authentication".)
- No clear text passwords on the internet or service network. (Implemented by administrator's use of SSH, and business application's use of HTTPS/SSL, authentication will be server side certificates, client side password (SSL encrypted over the internet))
- Availability should be ensured. (Implemented by bastion host web server, and quality of service contract with Internet Service Provider (ISP))
- Web server must be hardened to bastion host level. (Implemented by parsimonious installation, kernel settings, single service.)

To ease adoption by partners, suppliers and customers, GIAC Enterprises will buy a site certificate signed by Verisign. This allows all users to verify the chain of trust without manually importing a special GIAC self signed certificate and accepting its authenticity. (Verisign's Certificate Authority certificate is built-in to all Netscape browsers.)

2.6.2 Description of SSL

SSL is the means by which GIAC ensures confidentiality, integrity, and authentication. Confidentiality is provided through the use of server/client negotiated encryption algorithms. Integrity is accomplished through the use of server/client negotiated hash functions. Finally, authentication is accomplished from the server side by public key certificates, and on the client side, by strong passwords passed in encrypted form over the internet.

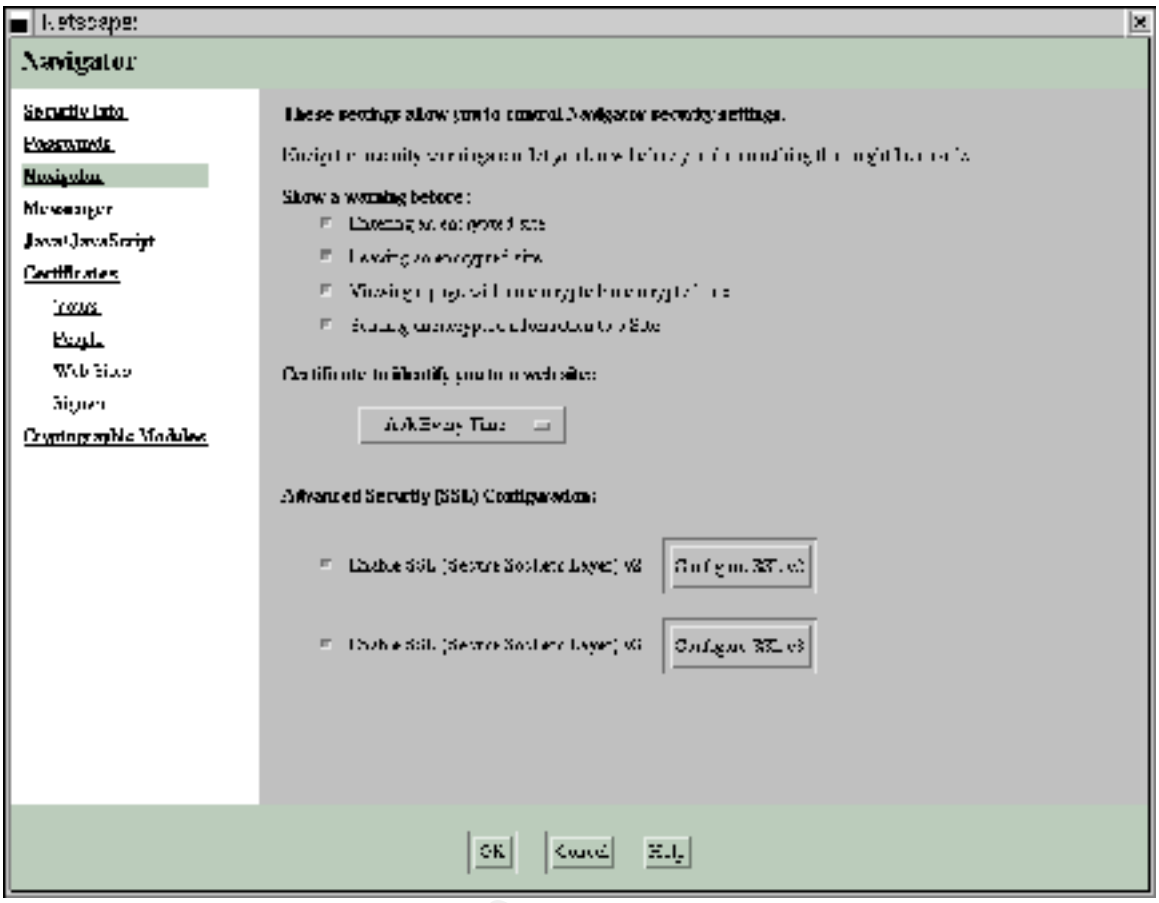
The web server has control over the strength of encryption and hash functions utilized in the transaction. It is configured to utilize only certain encryption levels and hash functions. When contacted by a client (which passes the encryption and hash levels that it supports), the sever chooses the highest common server/client cypher suit and hashing methods. The server will terminate the session if no commonality exists. So GIAC can control the necessary level of encryption and hashing by only supporting those that provide sufficient security.

2.6.3 Conceptual SSL Sequence of Events

✂Client opens a connection to the web server.
✂Server sends its public certificate.
✂Server and client negotiate cypher and hash to use.
Encryption of session begins.
✂Server requests password.
✂Client authenticates with strong password.

2.6.4 Browser Configuration

The browser configuration consists of enabling the desired SSL v2 and SSL v3.
This is done from the pop up screen shown by selecting the "security" button on the browser's icon bar. The proper settings correspond with the default installation settings and therefore require no additional set up effort. The pop up screen is shown below:



2.6.5 The HTTPS Server Configuration

The web server configuration is typically done by web browser access. The web server is accessed at an administration port (http://w.x.service.web:1234), which then displays the management controls as shown below:

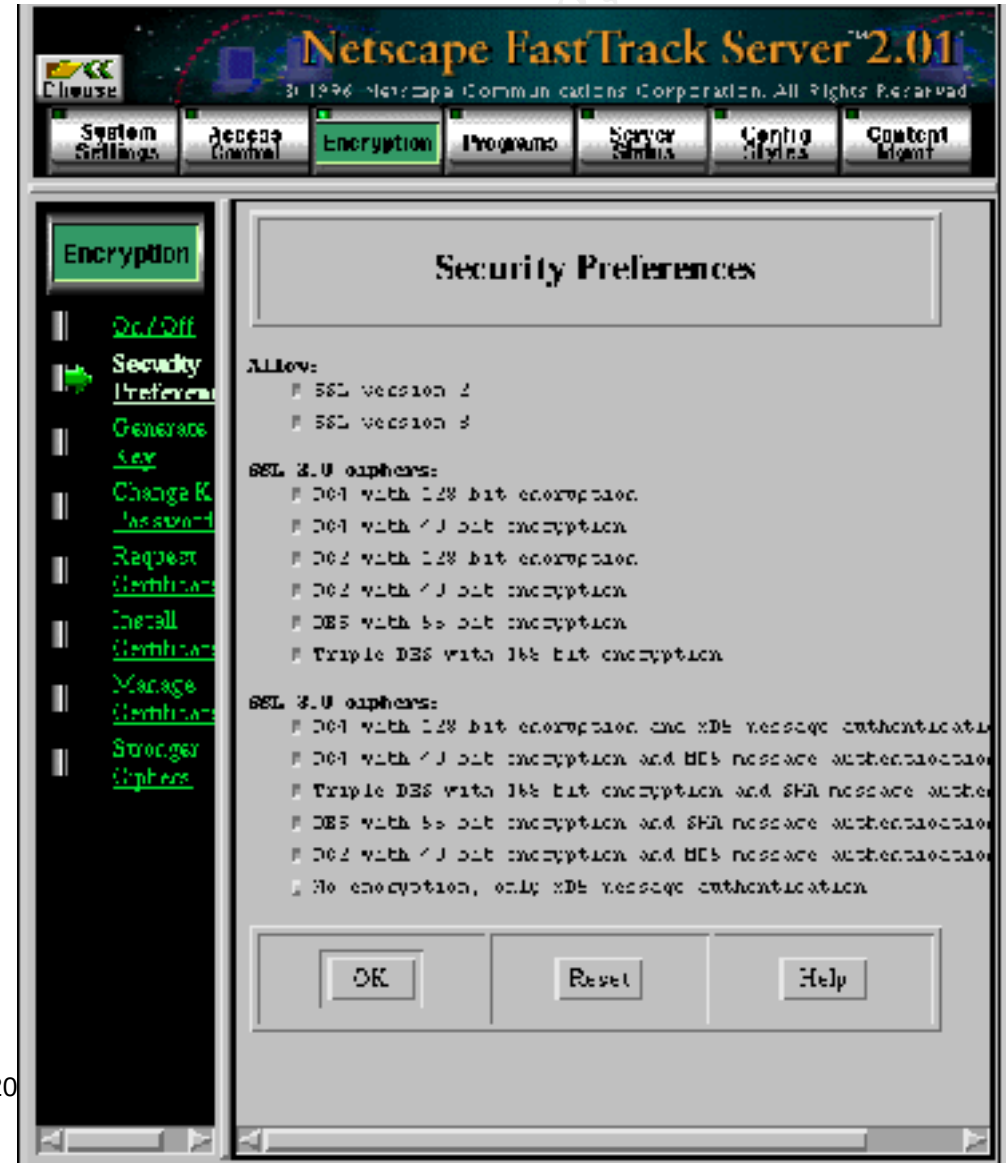
To set the web server's acceptable encryption and hash functions, select the "Encryption" button. The resulting default settings are shown below:

Note that these differ widely from our policy and will need to be set to only:

1. SSL 3.0 ciphers: RC4 with 128 bit encryption and MD5 message authentication.
2. SSL 3.0 ciphers: Triple DES with 168 bit encryption and SHA message authentication.

2.6.6 Procedure for Enabling SSL on a Netscape Server

- Generate a key pair.
  - As root, in server root dir ./bin/httpd/admin/bin/sec-key
  - Key pair file is in <server root>/httpd-<server-name>/config/ServerKey.db
  - Enter a password for key pair. Once SSL enabled, must type password every time server is restarted.
  - In Server Manager, select Encryption->Generate Key, type path and filename of



key file (relative to server root dir)

- Get certificate from Certificate Authority.
- Install the Certificate.
  - In Server Manager, select Encryption->Install Certificates.
  - Select "This Server", paste public key encrypted certificate into form.
  - Save and Apply.
- Turn on SSL.
  - In Server Manager, select Encryption->On/Off.
  - Select "On" button, type port 443.
  - Enter location (as relative path) of server's key file.
  - Enter location (as relative path) of server's certificate.
- Set security preferences.
  - In Server Manager, select Encryption->Security Preferences.
  - Select SSL versions to use: SSL 3.0.
  - Select cipher and hash combinations: "RC4 with 128 encryption and MD5 message authentication" and "3DES with 168 bit encryption and SHA message authentication".

2.6.7 VPN Test

Testing that the VPN conforms to the security policy consists of verifying the following specifics of the implementation:

- Verify that Quality Of Service (QOS) relationship exists with the ISP.
- Verify that the web server settings that it offers and negotiates with clients, is limited to the two choices of triple DES or RC4 with 128 bit encryption.
- Verify that backup and maintenance cron jobs utilize SSH.
- Sniff the web server's SSL traffic and its Oracle traffic, checking the payload for readable data by using the tool "sniffit" (it shows the ASCII contents of the sniffed session).
- Ensure that no vulnerable Common Gateway Interface (CGI) scripts are installed on the web server by running the "whisker" tool . Whisker is a CGI vulnerability scanning tool which is meant to be run against web sites.
- Ensure bastion host level of configuration by vulnerability scanning with "nessus" tool. Verify the configuration settings in /etc/system to defeat stack buffer overrun attacks (set noexec_user_stack = 1, set noexec_user_stack_log = 1).

3.Assignment 3 - Audit Your Security Architecture - Primary Firewall

The following test provides a description of a comprehensive audit of the primary firewall. This firewall is implemented with Checkpoint's Firewall 1 V 4.x running on a Sun Solaris 8.x platform. GIAC is primarily a UNIX/Linux populated company, and administrator expertise exists for hardening this platform.

This firewall is meant to be the primary defense, and must provide the far greater security of stateful inspection. This means that policy allowed reply ports of provided services are only opened through the firewall when they are needed. For TCP, this means established connections. For UDP, this means internal initiated sessions with time outs of responses.

At this time, the NAT feature of FW-1 is not required or desired (will impact the use of ssh to administer and backup service network servers).

3.1 Considerations and Principles

The security policy defines the desired behavior. Any findings reflect the differences between the desired and the actual implementation.

- Audits should be periodic, since intranets and their firewalls are constantly evolving entities. It is recommended that audits be done biannually.
- Findings should feed back into the process to strengthen both the implementation and the process.
- Audit reports are sensitive! These generally reveal weaknesses in implementation.

3.2 Risks, Tasks and Estimated Costs

Note that this is an audit of the primary firewall, not a penetration test or black hat simulation. Its intent is to show that the FW-1 is in compliance with the formal specification of the security policy (See Assignment 2 for FW-1 policy specifics).

There are basically two types of risks to be considered, political and economic. From the political perspective it is necessary to have the written support of a high level GIAC executive. This provides the authorization to perform a thorough assessment of the primary firewall. Executive support is difficult to get because security does cost money, and is very difficult to estimate its dollar measurable return.

From the economic perspective, assessment of the firewall requires considerable effort and could result in system down time. To reduce this risk and to make the analysis of test data easier, the audit should be performed at a time of low business need and low network traffic. The first day of a three day weekend would be ideal. This would allow enough time to do partial retests if ambiguous results were discovered. It would also

allow recovery if a system were accidentally brought down.

Tasks, Level of Effort, Costs Table

Tasks	*Level of Effort*	*Costs*
Review security policy	½ person day	$ 0.25 k
Execute filter tests described in assignment 2, review resulting logs	1 person day	$ 0.5 k
Ensure bastion host configuration	1 person day	$ 0.5 k
Analyze and compile results	1 person day	$ 0.5 k
Generate report	½ person day	$ 0.25 k
Act to correct any findings, and to prepare for next biannual audit	1 person day	$ 0.5 k

3.3 Implementation of Assessment Specifics

3.3.1 Verify Rules

Test who can and who can't use which services of the GIAC resources behind the FW-1 firewall. This typically involves the use of "nmap" to verify the firewall rule set. Since GIAC incorporates heavy logging on its rules, the activity of the accept or deny of the stimulus can be verified conveniently here. Check the resulting logs for the expected logging messages.

See Tables of rules tests from **Assignment 2 - Security Policy, 2.5.7 Primary Firewall Rule Testing** for these specific procedures/tests.

3.3.2 Verify Bastion Host Level of Configuration

Test the configuration and the security of the firewall host itself. The FW-1 product is installed on a Sun Solaris 8.x system. There are several tools that can be used to ensure this level of security:

- ISS or Nessus: These tools are vulnerability scanners. They detect open services and look for the existence of known vulnerabilities. The successful passing of this scan provides you more assurance of a hardened system capable of withstanding attacks.
- Tripwire: A baseline tripwire run should have been run at build time of this platform, before it was even exposed to the network. At each audit, tripwire is rerun and compared to this baseline stored on write only media. This has two benefits. First you are ensuring that the configuration files of the firewall application have not been unknowingly modified. Second, you are ensuring that the basic operating system files of the hosting platform have not been modified unknowingly. Both together give you confidence of correct firewall configuration, and that the host has not been compromised.

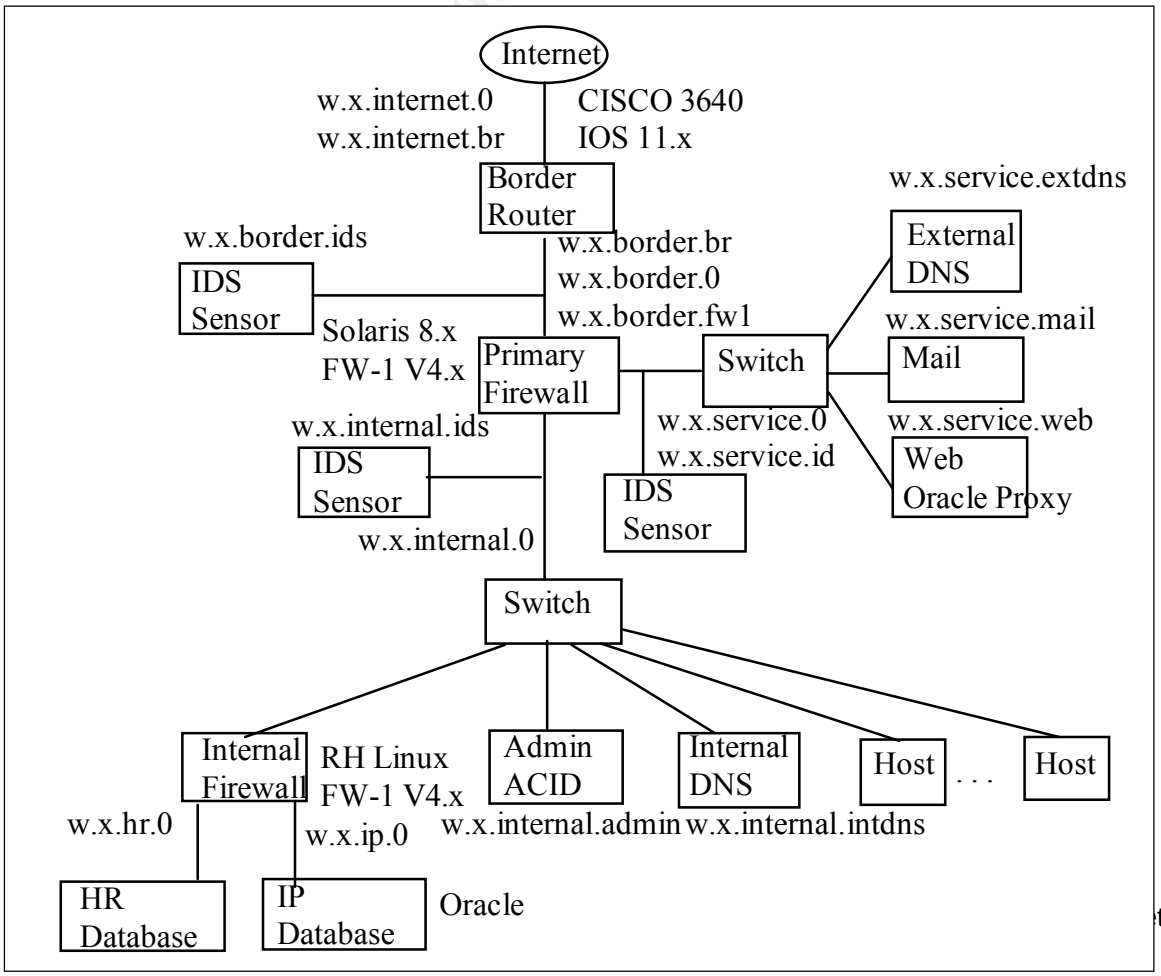
- Patches: Verify that the system is up to date on security patches.
- Accounts: Confirm that only administrator accounts exists. This should be a single service server with no users for highest security.
- Process Snapshot: use "ps -Af" to compare against a baseline snapshot taken at system build time.
- Netstat Snapshot: use "netstat -a" to compare against a baseline snapshot taken at system build time.
- Lsof Snapshot: use "lsof" to compare against a baseline snapshot taken at system build time. Use "lsof -i" to show all open ports associated with the currently running processes.
- Test correct fail conditions and ensure that no unwanted packet forwarding occurs, do each of the following:
    1. Simulate a power failure caused reboot.
    2. Execute a normal reboot.
    3. Kill the firewall daemon.
        kill -9 <firewall-1 daemon>

3.4 Perimeter Analysis Improvements/Alternate Architectures

After the assessment, the architecture seems sound. However, from ISS scan results of the web server and the large number of web based vulnerabilities being discovered, it seems prudent to further enhance the security of the web service. The most significant enhancement would be to evolve from using HTTPS client password authentication, to using one time password generating tokens (SecurID) for partner and suppliers. Note that GIAC has no control or knowledge of the security over these hosts. This would reduce the risk to GIAC from compromised partner or supplier hosts.

An additional improvement would be to replace the internal filtering firewall with a stateful one. The filtering approach seems a little weak considering the potential for damage. An insider with their GIAC specific knowledge combined with their internal network access, could generate a formidable attack. A stateful firewall of a different brand then FW-1 would be recommended. Perhaps even running it on a different operating system, say Linux, would add additional complexity to an attack. See the newly proposed architecture immediately below:

I am somewhat unconvinced of the utility of the IDS sensor between the border router and the FW-1. It generates a lot of noise but its findings consist mainly of low value stealth type scans.

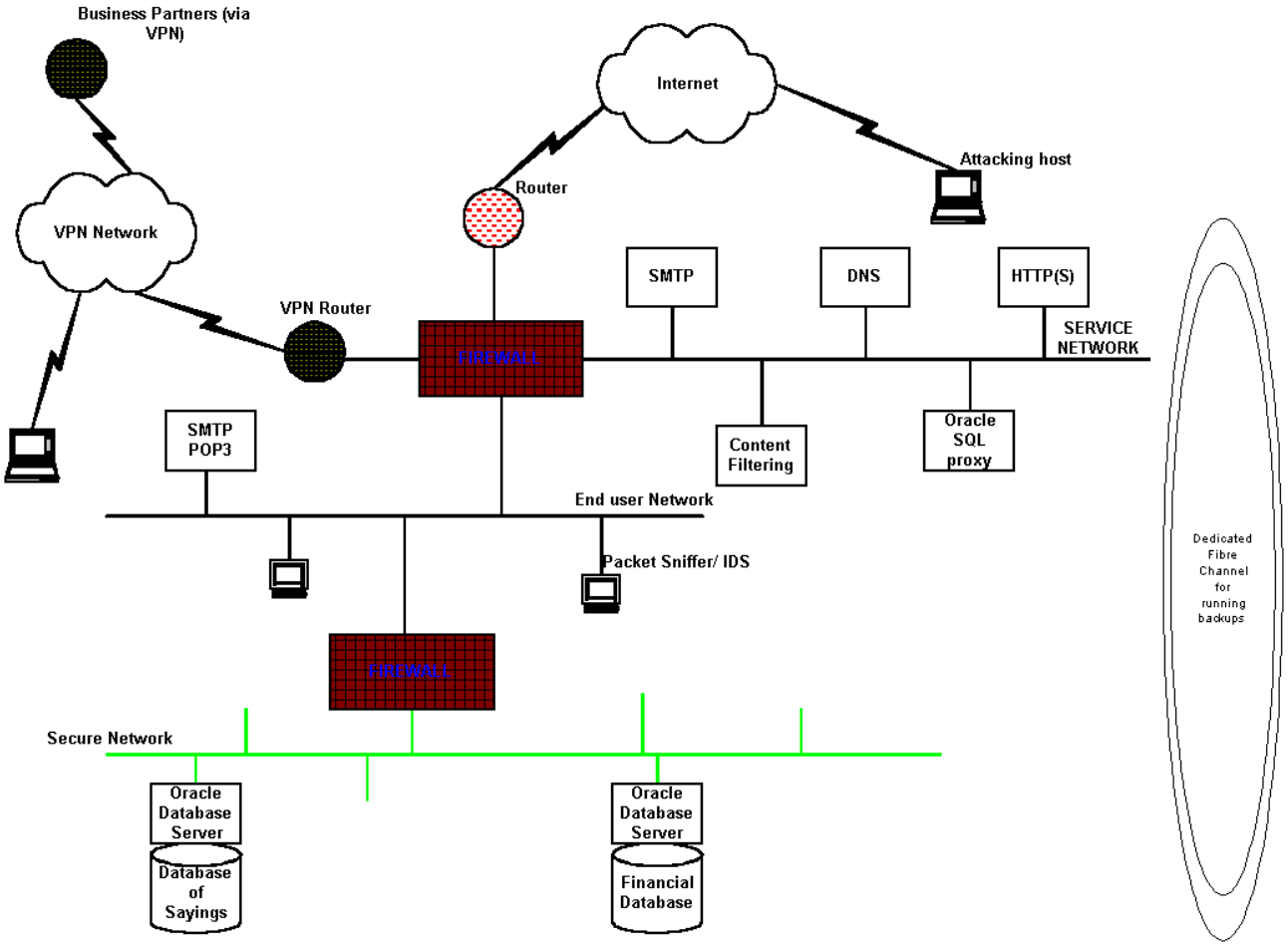


4. Assignment 4 - Design Under Fire

4.1 Background

The network design selected to attack is described by URL:
http://www.sans.org/y2k/practical/Kofi_Arthiabah.zip

Diagram from Kofi_Arthiabah.zip



Proposed GIAC Enterprises Network Security Architecture

4.1.2 The Attack

The attack description can be found at URL:
http://www.phoneboy.com/docs/bh2000.blackhat-fw1.html
The attack used is demonstrated and implemented by URL:
http://www.phoneboy.com/docs/bh2000/blackhat-fw1.tar.gz
See program: fw1none

This tar file actually contains a set of attacks, but the one of interest is an attempt to subvert the authentication process on the administrative control channel of the FireWall-1. In version 4.0 the administrative port 256 tcp is open to "any" and can be exploited with either of:

- A known IP address of a management module, or
- A common misconfiguration of the control.map file (which is usually done to make local administration easy to accomplish).

  127.0.0.1: */none

The intent of the attack is to bypass the usual authentication steps, allowing the attacker to issue an unload command to the firewall.

4.1.3 Reconnaissance Phase

To gather information for the attack, we use "whois" to determine GIAC's IP address block along with their mail and dns server addresses. We could also guess from nslookup results on the GIAC advertised public web server. We now have a good idea of the IP range to scan. We also know the specific addresses of the service network hosts (mail, dns, and web).

Next, a stealth scan using "nmap" is applied. What we are looking for is network structure and host information, particularly the FW-1 remote administration host. We should scan:

- The FW-1 host itself. (nmap will likely succeed here)
- The service network. (nmap will likely partially succeed here, with information inhibited by the FW-1's rules.)
- Any internal network. (nmap may not yield much here, due to restrictive FW-1 rules)

The tool "firewalk" can be employed to yield information about services allowed through the FW-1 deployment.

4.1.4 Attempt the Attack

Try the attack directly (using a spoofed 127.0.0.1 source IP), hoping for the 127.0.0.1 misconfiguration (and that the border router doesn't filter it out in its unroutable ACLs). If indications are that the border router is filtering on source addresses of 127.0.0.1, we can try to encapsulate the message in FWZ. This encapsulation technique can be used to send packets through the firewall, avoiding the filtering or stateful inspection phase.

If our reconnaissance has indicated some likely candidates for the remote administration host, try the attack using those as spoofed source addresses. If this fails, we can brute force scan for the correct administration IP by just trying each suspected IP in GIAC's address block.

4.1.5 Alternate Approach

Maybe the defense in depth combination of the border router and the internet inbound rules of the FW-1 are too strong. We can attempt to compromise a service network host, and continue the same basic attack against the FW-1, but from the service network interface, hoping for a looser set of rules.

To attempt this, the best bet is to use the CGI vulnerability scanner "whisker" against the web server. Whisker incorporates several evasion techniques which help it to penetrate firewalls that may do some data payload scanning. In particular, it has the ability to fragment the packets of its scan attempts, and it uses various unicode representations to avoid detection and blocking. If any root acquiring vulnerabilities are reported , we apply the corresponding exploit to gain access to the host. Once established on the service network, we reapply the approaches of 4.2.2 and continue the attack.

If the web server is too tight, we then try the general vulnerability scanner "nessus". It not only knows about the web service, but also is aware of many operating system specific service vulnerabilities. It should be used to scan every host on the service network (web, dns, mail, and the firewall platform itself). Note that nessus is extremely heavy handed and noisy, but effective. We then elect the most likely discovered vulnerability, apply its exploit, and use that host as the base for continuing attacks on the FW-1 host.

4.2 A Denial of Service Attack

Given that GIAC may be subjected to massive TCP SYN, UDP, or ICMP floods from as many as 50 cable modem/DSL systems, what can be done? There are three main approaches to help reduce the effects of a massive DoS attack, whether it is TCP SYN, UDP, or ICMP.

- Process them "smarter" at the destination host.
- Don't allow them to reach the destination.
- Have alternate internet access (redundancy).

4.2.1 Better Destination Handling

Handling them better at the destination typically means applying all related patches to potentially exposed systems. Operating system vendors have recognized the TCP SYN flood problem and modified parts of the TCP/IP stack along with associated time out values to tune the system, reducing the paralyzing effects of the attack.

4.2.2 Filter Before Destination

There are several approaches to denying DoS attacks ability to reach their target destination. The first is to use IDS to detect quickly the problem (the source IPs of the flooding hosts) and then act on protecting the network by applying the specific host address filters at the border router. These routers, if properly configured, should already be blocking UDP and ICMP from general internet sources. If you have a quality of service relationship with your ISP, then they are obligated to respond with filtering of

their own at an upstream point. And finally, there is the option of fronting your border router with a traffic shaping appliance. This works by dropping excess traffic of the specified types that exceed allowed traffic thresholds. It is applied as a selected traffic throttling mechanism.

4.2.3 Alternative ISP Accessibility

Having alternate, redundant, internet access can help. This means having several ISPs which supply your internet access. The theory being that the DoS attack will likely be coming through one or the other ISP gateways, so if worse comes to worse simply disconnect from that DoS source.

4.3 Internal System Compromise Process

The discussion below assumes that the attack of 4.1 fails. If 4.1 works, then we have successfully executed an "unload" command on the firewall, and any attack would proceed through that compromised firewall as the path of least resistance.

The selected internal target is the Oracle financial server. It contains the customer transactions with the company. This credit card information is of high value. The financial server is also more exposed to attack for several reasons. One, it must have a means to communicate with the web server on the service network. This communication means that holes have been made in the FW-1 firewall to allow for it. But the web server can be reached by anyone on the internet, which makes it more vulnerable to internet attacks. Secondly, databases historically have implemented security measures and features in an add-hoc add-on fashion. They are generally fairly weak. So it is believed that the most likely attack should be made initially against the web server, which is then used as a stepping stone to the financial server. See 4.1.5 Alternate Approach, for the plan to compromise the web server. Normally this is accomplished through a buffer overrun based attack.

Once root has been achieved on the web server, the plan is to access the various databases in the financial server by mimicking the transaction process normally executed on the web server. We could even use the web server's /bin/cgi programs by calling them directly, but supplying our own arguments. It is anticipated that this level of compromise may be sufficient to gain the desired credit card information.

But what if it isn't? We may have to break into the database server host itself. This means using application level Oracle exploits to trick the server into executing a shell instruction sequence which allows us dome degree of access as a host user. At this point, though, we would still have to overcome the potentially constraining rules of the internal firewall.

So we are back at the process of reconnaissance. We must find a way to bring in some tools to the web server. This could be done by tunneling them inside the HTTP protocol (remember we now control the web server). As usual, nmap and firewalk can be applied

against the internal firewall and the network behind it.

This reconnaissance, vulnerability scan, exploit cycle is reapplied in an attempt to work our way deeper into the intranet.

List of References

1)Venema, Wietse and Dan Farmer. "Security Auditing and Risk Analysis". 1996.
2)Goncalves, Marcus and Steven Brown. "Check Point FireWall-1 Administration Guide". New York: McGraw-Hill Book Co., Inc., 2000
3)Brenton, Chris and Gary Bibeau, Stephen Northcutt, Christoher Pettit, Char Sample, Sean Schwoerer. "Firewalls 101: Perimeter Protection with Firewalls". The SANS Institute, 2000.
4)Brenton, Chris and Stephen Northcutt, Lance Spitzner, S. Winters. "Advanced Perimeter Protection and Defense In-Depth". The SANS Institute, 2000.
5)Anon. "VPNs and Remote Access". The SANS Institute, 2000.
6)Anon. "Network Design and Performance". The SANS Institute, 2000.
7)Ritchey, Paul. "Introduction To Snort"
8)Northcutt, Stephen. "Network Instrusion Detection An Analyst's Handbook". Indianapolis, Indiana: New Riders Publishing, 1999.
9)Cole, Eric and Edward Skoudis. "Computer and Network Hacker Exploits, Part 1, 2, &3". The SANS Institues, 2001.
10)Apple, Chris and Mike Barbarino, ..., Aruna Victor. "Netscape FastTrack Server Administrator's Guide Unix", NetscapeCommunication Corp. 1995.

URL References:

Http://www.snort.org: for the "snort" network IDS.
Http://www.nessus.org: for the generic vulnerabilities scanner "nessus".
Http://www.wiretrip.net/rfp: for the CGI vulnerabilities scanner "whisker".
Http://www.phoneboy.com/docs/bh2000/blackhat-fw1.html: for FW-1 vulnerabilities.
Http://www.phoneboy.com/docs/bh2000/blackhat-fw1.tar.gz: for the exploit demo code.
Http://www.packetstorm.security.com/UNIX/audit/firewalk: for the scanning of firewall "through ports" scanner "firewalk".
Http://www.sans.org/y2k/practical/Kofi_Arthiabah.zip: for the network diagram of the attacked design.
Http://www.cert.org/kb/acid: for the Analysis Console for Intrusion Databases (ACID).