GIAC GCFW Practical Assignment v. 1.9

Alexandre da Costa Medeiros

September 2003

Table of contents

Abstract

This paper defines the network security architecture for GIAC Enterprises, a fortune cookie sayings company. The first part of this document describes the business operations, network layout and access requirements for customers, suppliers,

partners and employees. Also is described the requirements and placement of essential components such as the border router, primary and internal firewalls, VPN server, network intrusion detection sensors and all public servers. In the second part of this document, it is define the security policy for the border router, primary and internal firewalls. In the third part, it is presented the result of a complete audit of primary and internal firewalls. The purpose of this audit is to verify if the security policy is implemented and to make a few considerations and recomendations. In the last part, it is presented an attack strategy for a previous security design of GIAC Entreprises and it demonstrates how this architecture could be compromised today.

Assignment 1 - Security Architecture

1.1 - Introduction

GIAC Enterprises is an e-business company which sells fortune cookie sayings, acquired from writers who lives in different places in the world. Customers of GIAC Enterprises can purchase sayings accessing main GIAC's website http://www.giacfortunecookies.com. All fortune cookies are categorized by subject, such as birthday, party, love, etc. The company has 32 employees, 16 reseller partners and 16 suppliers. The revenue of GIAC Enterprises is about five million american Dollars, and it is vital for the company that all transactions must be done securely. To achieve this level of security, the infrastructure must be secured.

The purpose of this document is to present the necessary infrastructure to ensure GIAC Enterprises is properly secured to operate on the Internet.

1.2 - Scope

It is scope of this document discuss the IP address scheme, placement of essential components, such as the border router, firewalls and VPN server. It is necessary to include the brand, version, description of each listed component. It isn't scope of this document describe the communications infrastructure such as telephone and fax.

1.3 - Access Requirements and Restrictions

In this section, is defined the access requirements and restrictions for Customers, Suppliers, Partners, Employees and Mobile taskforce and Teleworkers.

1.3.1 - Customers

The main communication channel between customers and GIAC Enterprises is the website www.giacfortunecookies.com. It is possible browse the different types of catalogs and samples. Sayings are grouped by subject such as birthday, pary, love, etc. At any time, customers can add to their shopping cart, a package of sayings.

At a convenient time, customers may finish shopping clicking on a specific link called ``Procede to CheckOut'' at the top of the page. At this moment, it will be redirected to another site, buy.giacfortunecookies.com, where only HTTPS is supported, allowing a secure enviroment to finish the comercial transaction.

Once the transaction ins concluded, customers will have access to the sayings package, clicking on ``Download Sayings" link. A compressed file with the .zip extension will be sent to the customer.

Customers may use other forms of contact with GIAC Enterprises such as, e-mail, Telephone and Fax.

For e-mail contact, customers can send e-mail for two addresses: info@giacfortunecookies.com and sales@giacfortunecookies.com. Both are used for general information and sales information, respectively. E-mail is a naturaly insecure medium, no customer sensitive information is transmitted by e-mail. If necessary, customers may contact via Telephone or Fax.

The privacy policy is available at the main page of the website www.giacfortunecookies.com and may be viewed, accessing the link ``Privacy Policy" in the main page.

1.3.2 - Suppliers

Suppliers of GIAC Enterprises are writers who lives in different parts of the world. The contact form with GIAC Enterprises are essentially the same used by customers, they can access the website, and contact GIAC via e-mail, Telephone or Fax.

Their work could be submitted via a SSL-protected form, encrypted e-mail or Fax. GIAC Enterprises reserves the right to no accept works by Telephone.

To submit their work via a protected form, suppliers have to click on the link ``Partners" in the main page of www.giacfortunecookies.com. Doing this, will redirect to another site, suppliers.giacfortunecookies.com. This website supports only HTTPS and suppliers must have a X.509 v.3 certificate to have access to their individual enviroment. X509 certificates may be issued by a 3rd party such as Verisign or GIAC Enterprises, using OpenSSL.

Once authenticated, the supplier have access to the options ``Submit Work", ``Work Submition History", ``Payment History" and ``Log Out".

If the supplier wants or need to send his work via e-mail, it will be oriented to do so securely, using PGP to encrypt data. GIAC Enterprises reserves the right to not accept unencrypted work via e-mail.

The last option to submit their work is via Fax, but GIAC Enterprises defines by contract that all works submitted by fax have a 75% of the price of work sent by protected form or encrypted e-mail.

1.3.3 - Partners

Partners are other companies that acquire sayings from GIAC Enterprises, translate to a foreing language and resell those sayings. Essentially, partners are considered special clients and due to that characteristics, partners has special enviroment to control their operations.

To access this enviroment, partners have to click on link ``partners" on main page of GIAC's website or access directly the partners.giacfortunecookies.com site.

This site supports only HTTPS, and in order to have access to the dedicated enviroment, partners must have a X.509 v.3 certificate. Once authenticated, partners have the same options as customers plus tools like advanced search, complete view of sayings packages and rank of best sellers.

Partners, like regular customers, after choosing sayings packages, may finish the transaction clicking on ``Procede to Checkout" link at the top of the page. Once the transaction is concluded, partners receive the packages same way regular customers do, a compressed file with .ZIP extension which contains one or more files with sayings.

Contact may be done by e-mail, but only non sensitive information is transmitted with this media. The site partners.giacfortunecookies.com was developed with the objective to have all necessary functionalities for partners operations. Encrypted with PGP is possible but isn't encouraged by GIAC Enterprises.

Contact by Telephone or Fax can be done, but restricted to non sensitive information, like payment confirmation and info on site usage. GIAC Enterprises reserves the right to not send packages via e-mail, Fax or Telephone.

1.3.4 - Employees

GIAC Enterprises HQ is placed in the city of Campinas, state of São Paulo, Brasil. To develop its activities, the employees have different access level on services and applications. Depending on employee's function, a higher or no access to a specific service or application is allowed.

Essentially, internal users only have access to the Internet via application proxies. Direct access is allowed to system administrators and security analysts, for administrative purposes only.

When internal users wants to query a public name in DNS, the query is first sent to internal DNS resolver, graviola.giacfortunecookies.com, that will query other DNS servers outsite GIAC's LAN in order to provide the answer to the DNS client.

All message traffic from the Internet to GIAC Enterprises, are received by the SMTP servers a.mx and b.mx, both placed in the DMZ. These servers are configured to accept e-mail only for the domain giacfortunecooikes.com. Some anti-SPAM countermeasures are configured, such as filters to block e-mails form Open Relays and Dial-up User List entries listed in MAPS's database. Every message is also checked by a anti-virus, and attachments with the extensions .exe, .pif. .src are prohibited. Other file extensions such ass .gz, .tar, .tar.gz, .tgz, rar, .zip, .gif, .pdf, .jpg, .png, among others are accepted.

After checking the messages for bad senders and viruses, they're forwarded to a internal server, a-int.mx.giacfortunecookies.com, where internal users can get their messages via imap or pop3 (both with SSL support). Messages to outside GIAC Enterprises, do the reverse path described above.

In order to browse the web and access multimedia streaming services, users must use a proxy that wil retrieve the content or page for the client. Users must authenticate with the proxy server, before accessing a external website or multimedia stream. Certain file types and site locations are prohibited for download and access.

The use of proxy may help to save bandwidth usage when two or more users access the same information, and could be used to log and monitor access to websites.

1.3.5 - Mobile Sales Force and Teleworkers

Some employees work outside of the office and consequently does not have direct access to the GIAC Enterprises LAN. These employees are the salespeople and teleworkers. Both only have access to the LAN via a VPN server. First, it is necessary to establish a connection with a local ISP before authenticate with the VPN server. A VPN client software is necessary to authenticate with the VPN server and access the internal LAN of GIAC Enterprises.

The software bundle used by Salespeople and Teleworkers includes Microsoft Windows XP, Microsoft Office XP, SSH Sentinel 1.4 and Norton Antivirus 2003.

1.4 - Architecture

The objective of the security architecture, is to implement security in depth, using different defenses in the local network. The idea is to have as many as possible layers of defense, so in case one or more defenses are compromised, it does not mean the entire network was compromised.

Other aspects considered in the design of the security architecture:

- Monitoring and managing servers and devices - Considered essential on identifying attacks and problems of GIAC Enterprises LAN. The intelligent use of network-based IDSs, SNMP and Cisco's NetFlow, contribute to enhance the overall security of the network. It is necessary the knowledge to interpret the data collected, in order to make the correct decisions;

- Expandability - If it is necessary an increase on processing or network capacity, it is not necessary rebuild the entire architecture, in order to handle the load;

The security architecture for GIAC Enterprises enforces the separation of the LAN in different groups through the use of dedicated switches and packet filtering on firewalls inside GIAC's LAN.

Firewalls used in this architecture are based on the GNU/Linux and FreeBSD operating systems. It is used different technologies because if one problem is encountered one system, it is unlike to appear on the other. In spite of increase of complexity, using different technologies brings robustness to the architecture.

Network-based IDS sensors are placed on all segments of the LAN, to monitor and detect known malicious activity. All information collected by the sensors are stored on a database server for further analysis.

SNMP is used to register the health of server and devices of GIAC's network and NetFlow is used to register the different flows incoming and outgoing GIAC Enterprises. It uses flow tools, like FlowScan to summarize and make graphics of network usage. It is a very important set of tools on detecting and tracking **Denial of Service** attacks.

Details on each component used in the GIAC Enterprises security architecture are found on section Network Components, on section 1.8.

1.5 - Connection to ISP

GIAC Enterprises has a 4MBit/s connection with its ISP but it is considering a upgrade to 10MBit/s in the begining of the next fiscal year, in case the demand for fortune cookie sayings keeps increasing.

1.6 - IP Address Scheme

In this paper, the netblock 172.16.0.0/12 are treated as global routable prefix. GIAC Enterprises received from its ISP the 172.16.1.0/24 prefix for its exclusive use.

Moreover, GIAC Enterprises uses the private block 192.168.0.0/16 to organize the GIAC's LAN. The table below shows how are defined the different subnets in GIAC's LAN.

GIAC address space	172.16.1.0/24
DMZ segment	172.16.1.0/27
Web cluster segment	172.16.1.32/27
Intermediary (between firewalls) segment	172.16.1.240/29
Perimeter segment	172.16.1.248/29
Management/Services segment	192.168.1.0/24
Internal users segment	192.168.2.0/24
Database/Applications segment	192.168.3.0/24
IDS segment	192.168.255.0/24

Table 1 - IP address schema

1.7 - Network Diagram

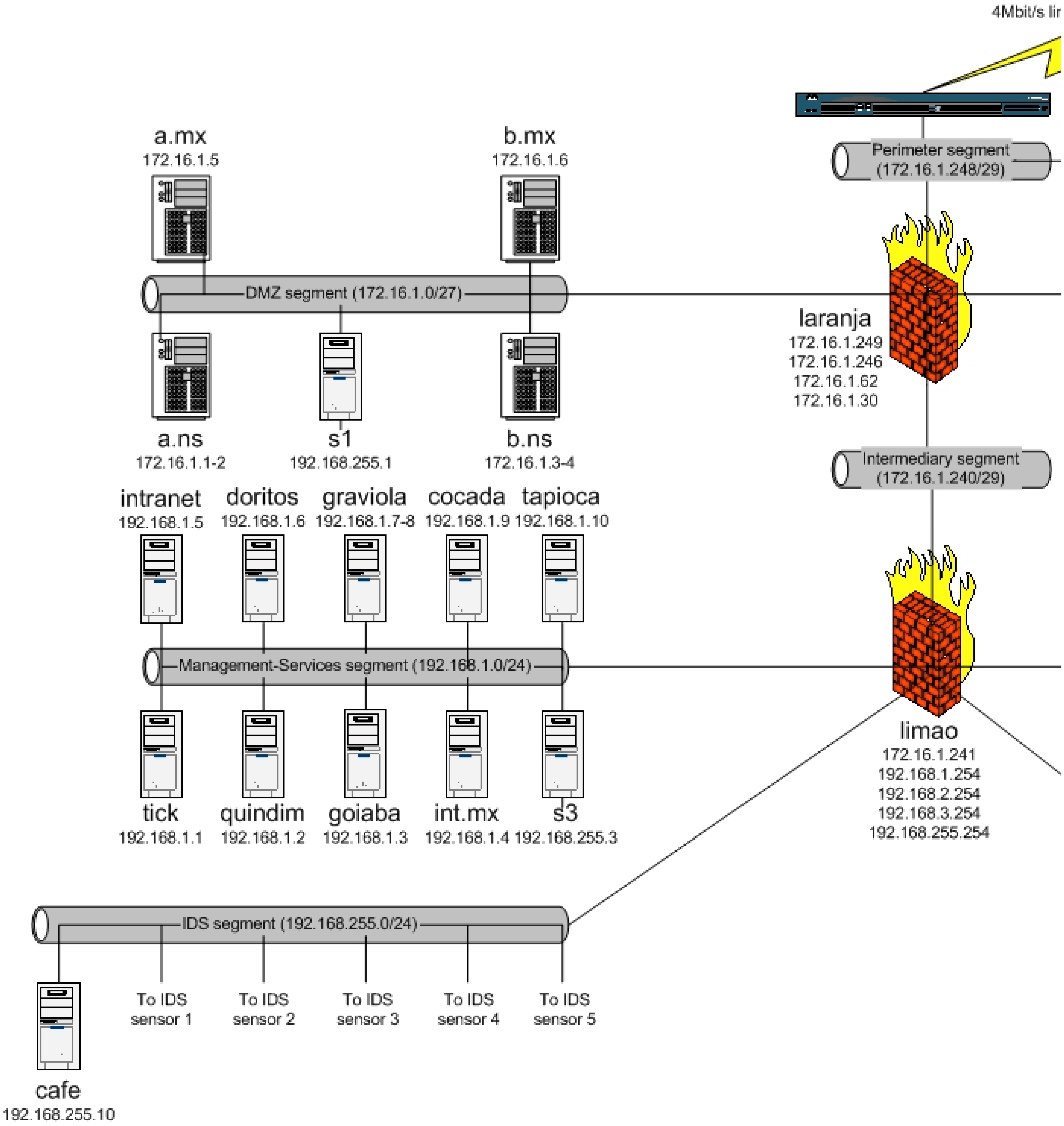


Figure 1 - Network Diagram

1.8 - Network Components

The components used in the design were chosen based on the best possible cost x benefit relation and on budget available. It was considered in the strategic business plan, an increase of 10%/year for the next period of three years.

Open Source software had the preference in the process of design of the network. This includes operating systems, backend and front end applications. Red Hat GNU/Linux were chosen to be used on backend servers and Windows XP for internal users, teleworkers and salespeople desktops and notebooks.

Red Hat Network Enterprise services was contracted to help managing system upgrades and package installations on servers using Red Hat GNU/Linux operating systems. CVSUP is used to upgrade all FreeBSD-based servers, when necessary.

SNMP is used on every device or server, in order to monitor the system's health, such as interface, memory and CPU utilization, etc.

Communications requirements are described for every component listed below. These tables will be useful when creating the border router, primary and internal firewalls policy on assignment 2.

1.8.1 - Border Router

- Name(s) / IP address(es): abacaxi.giacfortunecookies.com / a.b.c.d (external interface) and 172.16.1.254 (internal interface)
- Hardware: Cisco 3620 with NM-4E module
- Software: IOS 12.3(01)
- Function: Border router, the very first line of defense of GIAC Enterprises and gateway to the Internet
- Placement: Perimeter segment

Note: The Cisco 3620 plataform was chosen due its good packet routing capacity and cost x benefit relation. As the first line of defense it is configured on the router anti-spoofing rules, blocking all incoming and outgoing packets with source or destination IP addresses considered private, reserved, not allocated or bogus. It is also ensured by the border router, that all packets with destination to broadcast addresses are dropped, so GIAC's LAN cannot be used as amplification network.

Obs.: a.b.c.d is a IP address provided by the ISP.

application/service	protocol/port	accepts packets from	destination
SNMP	listening on 161/UDP	tapioca (src port >1023/UDP)	n/a
SYSLOG	uses src port >1023/UDP	n/a	514/UDP on goiaba
DNS client	uses src port 53/UDP or >1023/UDP	n/a	53/UDP on a.ns (172.16.1.2) or b.ns (172.16.1.4)
NTP	uses src port 123/UDP	n/a	123/UDP on tick

Table 2 - Services on abacaxi.giacfortunecookies.com

1.8.2 - Primary Firewall

- Name(s) / IP address(es): laranja.giacfortunecookies.com / 172.16.1.30, 172.16.1.62, 172.16.1.246 and 172.16.1.249
- Hardware: Dell PowerEdge 600SC (2.4Ghz Pentium 4, 512MB ECC 266Mhz DDR registered SDRAM, 40Gb IDE 7,200 RPM Hard Drive and 4 Intel Pro 1000/MT Ethernet cards
- Software: RedHat GNU/Linux 9.0 (updated on 06.28.03 12:00 EST), RedHat Network agent 1.0.2, IPTABLES 1.2.7a, Socklog 1.1.0, NET-SNMP 5.0.8 and Veritas NetBackup agent
- Function: Packet filtering, the second line of defense of GIAC Enterprises.
- Placement: Perimeter segment

Note: The combination Red Hat GNU/Linux + IPTABLES was chosen to be the base of the primary firewall of GIAC Enterprises. The Red Hat operating system has a efficient package management tool, and with the Red Hat Network Enterprise service, it is possible remote management of system upgrades and package installation, reducing downtime and work of the system administrator. IPTABLES is a modern packet filtering tool, extremely powerful. It supports complex protocols, stateful packet filtering (connection tracking), and has high packet troughput. If necessary, access can only be made using the console, the system does not accept remote terminal connections (SSH or Telnet).

application/service	protocol/port	accepts packets from	destination
SNMP	listening on 161/UDP	tapioca (src port >1023/UDP)	n/a
SYSLOG	uses src port >1023/UDP	n/a	514/UDP on goiaba
DNS client	uses src port 53/UDP or >1023/UDP	n/a	53/UDP on a.ns (172.16.1.2) or b.ns (172.16.1.4)
NTP	uses src port 123/UDP	n/a	123/UDP on tick
Red Hat Network agent	uses src port >1023/TCP	n/a	443/TCP on xmlrpc.rhn.redhat.com
Veritas NetBackup agent	uses src port >1023/TCP	n/a	6101/TCP on cocada
Veritas NetBackup agent	listening on 8192-3/TCP	cocada (src port >1023/TCP)	n/a
OpenSSH	listening on 22/TCP	192.168.1.0/24 (src port >1023/TCP)	n/a

Table 3 - Services on laranja.giacfortunecookies.com

1.8.3 - Internal Firewall

- Name(s) / IP address(es): limao.giacfortunecookies.com / 172.16.1.241, 192.168.1.254, 192.168.2.254, 192.168.3.254, 192.168.255.254

- Hardware: Dell PowerEdge 600SC (2.4Ghz Pentium 4, 512MB ECC 266Mhz DDR registered SDRAM, 40Gb IDE 7,200 RPM Hard Drive and 4 Intel Pro 1000/MT Ethernet cards
- Software: FreeBSD 4.8-STABLE (updated on 06.28.03 12:00 EST), IPFILTER v3.4.31 (336), Socklog 1.1.0 and NET-SNMP 5.0.8
- Function: Packet filtering, the third line of defense of GIAC Enterprises
- Placement: Middle of GIAC's LAN, it is connected with all internal segments

Note: The combination FreeBSD + IPFILTER was chosen to be the base of the internal firewall of GIAC Enterprises. It is considered a good idea to have different firewall technologies to protect the network. If one problem is encountered in one system, it is unlike to happen in the other. FreeBSD is a solid UNIX operating system, known for stability and high performance network capabilities. Although it does not have the same tools as Red Hat GNU/Linux, such as Red Hat Network, for system maintenence it has other tools that can be used for system upgrading such as CVS and CVSUP. IPFILTER is a stateful packet filtering tool, and was chosen because of its maturity, performance and reliability.

application/service	protocol/port	accepts packets from	destination
SNMP	listening on 161/UDP	tapioca (src port >1023/UDP)	n/a
SYSLOG	uses src port >1023/UDP	n/a	514/UDP on goiaba
DNS client	uses src port 53/UDP or >1023/UDP	n/a	53/UDP on graviola
NTP	uses src port 123/UDP	n/a	123/UDP on tick
CVSUP	uses src port >1023/TCP	n/a	5999/TCP on cvsup.freebsd.org
OpenSSH	listening on 22/TCP	192.168.1.0/24 (src port >1023/TCP)	n/a

Table 4 - Services on limao.giacfortunecookies.com

1.8.4 - Loghost

- Name(s) / IP address(es): goiaba.giacfortunecookies.com / 192.168.1.3
- Hardware: Dell PowerEdge 650 (2.4Ghz Pentium 4, 512MB ECC 266MHz DDR SDRAM, 40Gb IDE 7,200 RPM Hard Drive and 1 Intel Pro+ Ethernet card
- Software: Red Hat GNU/Linux 9.0 (updated on 06.28.03 12:00 EST), Red Hat Network agent 1.0.2, socklog 1.1.0, daemontools 0.86, NET-SNMP 5.0.8 and Veritas NetBackup agent
- Function: Centralized log server
- Placement: Management-Services segment

Note: Socklog will be used in favor of syslogd. Socklog supports log rotations based on file size, so log partitions can be calculated properly (there's no big surprises). It also supports sortable logs, log event notification and logs can also be transmitted through network using a TCP connection (errors in log transmissions can be handled).

application/service	protocol/port	accepts packets from	destination
SNMP	listening on 161/UDP	tapioca (src port >1023/UDP)	n/a
SYSLOG	listening on 514/UDP	172.16.1.0/24, 192.168.1.0/24, 192.168.2.0/24, 192.168.3.0/24, and 192.168.255.0/24 (src port >1023/UDP)	n/a
DNS client	uses src port 53/UDP or >1023/UDP	n/a	53/UDP on graviola
NTP	uses src port 123/UDP	n/a	123/UDP on tick
Red Hat Network agent	uses src port >1023/TCP	n/a	443/TCP on xmlrpc.rhn.redhat.com
Veritas NetBackup agent	uses src port >1023/TCP	n/a	6101/TCP on cocada
Veritas NetBackup agent	listening on 8192-3/TCP	cocada (src port >1023/TCP)	n/a
OpenSSH	listening on 22/TCP	192.168.1.0/24 (src port >1023/TCP)	n/a

Table 5 - Services on goiaba.giacfortunecookies.com

1.8.5 - VPN Server

- Name(s) / IP address(es): rapadura.giacfortunecookies.com / 172.16.1.253, 192.168.2.253
- Hardware: Dell PowerEdge 600SC (2.4Ghz Pentium 4, 512MB ECC 266Mhz DDR registered SDRAM, 40Gb IDE 7,200 RPM Hard Drive and 2 Intel Pro+ Ethernet cards
- Software: Red Hat GNU/Linux 9.0 (updated on 06.28.03 12:00 EST), Red Hat Network agent 1.0.2, FreeS/WAN 2.02, NET-SNMP 5.0.8 and Veritas NetBackup agent
- Function: Secure communications for roadwarriors
- Placement: Perimeter segment

Note: FreeS/WAN is a GNU/Linux implementation of the IPSEC protocol. It's being used to secure communications between GIAC Enterprise LAN and roadwarriors (salespeople and teleworkers). FreeS/WAN has a good interoperability with many IPSEC clients and server implementations (isakmpd, Kame, McAfee VPN, MS Win2k/XP, SSH Sentinel, etc.). Due reduzed number of roadwarriors, and actual speed of GIAC's link with its ISP, it is not considered an issue using a software-only VPN solution.

application/service	protocol/port	accepts packets from	destination
SNMP	listening on 161/UDP	tapioca (src port >1023/UDP)	n/a
SYSLOG	uses src port >1023/UDP	n/a	514/UDP on goiaba
DNS client	uses src port 53/UDP or >1023/UDP	n/a	53/UDP on a.ns (172.16.1.2) or b.ns (172.16.1.4)
NTP	uses src port 123/UDP	n/a	123/UDP on tick
Red Hat Network agent	uses src port >1023/TCP	n/a	443/TCP on xmlrpc.rhn.redhat.com
Veritas NetBackup agent	uses src port >1023/TCP	n/a	6101/TCP on cocada
Veritas NetBackup agent	listening on 8192-3/TCP	cocada (src port >1023/TCP)	n/a
IPSEC	listening protocol AH	Internet	n/a
IPSEC	listening protocol ESP	Internet	n/a
IPSEC	listening on 500/UDP	Internet	n/a
OpenSSH	listening on 22/TCP	192.168.1.0/24 (src port >1023/TCP)	n/a

Table 6 - Services on rapadura.giacfortunecookies.com

1.8.6 - Network Intrusion Detection sensors

- Name(s) / IP address(es): s1, s2, s3, s4 and s5 / 0.0.0.0 (s1-5) and 192.168.255.1-5
- Hardware: Dell PowerEdge 650(2.4Ghz Pentium 4, 256MB ECC 266Mhz DDR SDRAM, 40GB IDE 7,200 RPM Hard Drive and 2 Intel Pro+ Ethernet cards
- Software: Red Hat GNU/Linux 9.0 (updated on 06.28.03 12:00 EST), Red Hat Network agent 1.0.2, Snort 2.0
- Function: Monitoring the network, looking for malicious activity
- Placement: Every segment, except the perimeter segment

Note: Snort is a very good patern-matching IDS. One sensor is placed on every segment, that will look for known malicious activity. Every sensor has two ethernet interfaces, one is set at promiscuous mode without IP address it will be connected to a mirrored-port of the segment and the other is connected on a separated segment called IDS segment. The IDS segment isn't connected with other segments of GIAC's LAN. Logs are stored on a remotre SQL database for further analysis.

application/service	protocol/port	accepts packets from	destination
SYSLOG	uses src port >1023/UDP	n/a	514/UDP on goiaba
DNS client	uses src port 53/UDP or >1023/UDP	n/a	53/UDP on graviola
NTP	uses src port 123/UDP	n/a	123/UDP on tick
Red Hat Network agent	uses src port >1023/TCP	n/a	443/TCP on xmlrpc.rhn.redhat.com
Snort	listening on 0.0.0.0	n/a	3306/TCP on cafe
OpenSSH	listening on 22/TCP	192.168.255.0/24 (src port >1023/TCP)	n/a

Table 7 - Services on s1, s2, s3, s4 and s5

1.8.7 - External Domain Name System (DNS) servers

- Name(s) / IP address(es): a.ns.giacfortunecookies.com, b.ns.giacfortunecookies.com / 172.16.1.1-2, 172.16.1.3-4
- Hardware: Dell PowerEdge 650 (2.4Ghz Pentium 4, 512MB ECC 266MHz DDR SDRAM, 40Gb IDE 7,200 RPM Hard Drive and 1 Intel Pro+ Ethernet card
- Software: Red Hat GNU/Linux 9.0 (updated on 06.28.03 12:00 EST), daemontools 0.76, ucspi-tcp 0.88, djbdns 1.05, Red Hat Network agent 1.0.2, NET-SNMP 5.0.8 and Veritas NetBackup agent
- Function: Authoritative DNS for public names of giacfortunecookies.com
- Placement: DMZ segment

Note: Both servers are placed at the DMZ segment and only answers queries for its public domain giacfortunecookies.com. djbdns is a rock-solid DNS implementation, without known local and remote vulnerabilities since its birth. Zone transfers, between servers, will use rsync over a SSH tunnel. These servers will serve only public names, which includes the border router, primary firewall, VPN server, all servers at DMZ and Web Cluster segments. All internal names will be served by the internal DNS server.

application/service	protocol/port	accepts packets from	destination
SNMP (172.16.1.1 and 172.16.1.3)	listening on 161/UDP	tapioca (src port >1023/UDP)	n/a
SYSLOG	uses src port >1023/UDP	n/a	514/UDP on goiaba

Red Hat Network agent (172.16.1.1 and 172.16.1.3)	uses src port >1023/TCP	n/a	443/TCP on xmlrpc.rhn.redhat.com
NTP	uses src port 123/UDP	n/a	123/UDP on tick
DNS server (172.16.1.1 and 172.16.1.3)	listening on 53/UDP	Internet, DMZ and internal DNS resolver (src port 53/UDP or >1023/UDP)	n/a
DNS cache/resolver (172.16.1.2 and 172.16.1.4)	listening on 53/UDP	Perimeter, DMZ and Web cluster segments (src port 53/UDP or >1023/UDP)	n/a
DNS cache/resolver (172.16.1.2 and 172.16.1.4)	uses src port >1023/UDP	n/a	53/UDP on DNS servers outside GIAC's LAN
Veritas NetBackup agent	uses src port >1023/TCP	n/a	6101/TCP on cocada
Veritas NetBackup agent	listening on 8192-3/TCP	cocada (src port >1023/TCP)	n/a
OpenSSH	listening on 22/TCP	192.168.1.0/24 (src port >1023/TCP)	n/a

Table 8 - Services on a.ns.giacfortunecookies.com and b.ns.giacfortunecookies.com

1.8.8 - Internal Domain Name System (DNS) server/cache resolver

- Name(s) / IP address(es): graviola.giacfortunecookies.com / 192.168.1.7-8
- Hardware: Dell PowerEdge 650 (2.4Ghz Pentium 4, 512MB ECC 266MHz DDR SDRAM, 40Gb IDE 7,200 RPM Hard Drive and 1 Intel Pro+ Ethernet card
- Software: Red Hat GNU/Linux 9.0 (updated on 06.28.03 12:00 EST), daemontools 0.76, ucspi-tcp 0.88, djbdns 1.05, Red Hat Network agent 1.0.2, NET-SNMP 5.0.8 and Veritas NetBackup agent
- Function: Authoritative DNS server for internal names of giacfortunecookies.com and recursive DNS resolver for every server, device and workstation of GIAC's LAN
- Placement: Management-Services segment

Note: This server will answer recursive queries for all internal servers, devices, and employee's workstations using djbdns's dnscache. In the same machine, but different IP address, tinydns (part of djbdns package) will be used to serve all internal names. dnscache will be configured to query tinydns when necessary.

application/service	protocol/port	accepts packets from	destination

SNMP (192.168.1.7)	listening on 161/UDP	tapioca (src port >1023/UDP)	n/a
SYSLOG (192.168.1.7)	uses src port >1023/UDP	n/a	514/UDP on goiaba
Red Hat Network agent (192.168.1.7)	uses src port >1023/TCP	n/a	443/TCP on xmlrpc.rhn.redhat.com
NTP	uses src port 123/UDP	n/a	123/UDP on tick
DNS cache/resolver (192.168.1.7)	listening on 53/UDP	192.168.1.0/24, 192.168.3.0/24, 192.168.4.0/24 and 192.168.155.0/24 (src port 53/UDP or >1023/UDP)	n/a
DNS cache/resolver (192.168.1.7)	uses src port >1023/UDP	n/a	53/UDP on DNS servers outside GIAC's LAN
DNS server (192.168.1.8)	listening on 53/UDP	192.168.1.0/24, 192.168.2.0/24, 192.168.3.0/24 and 192.168.255.0/24 (src port >1023/UDP)	n/a
Veritas NetBackup agent	uses src port >1023/TCP	n/a	6101/TCP on cocada
Veritas NetBackup agent	listening on 8192-3/TCP	cocada (src port >1023/TCP)	n/a
OpenSSH	listening on 22/TCP	192.168.1.0/24 (src port >1023/TCP)	n/a

Table 9 - Services on graviola.giacfortunecookies.com

1.8.9 - External Simple Mail Transport Agent (SMTP) servers

- Name(s) / IP address(es): a.mx.giacfortunecookies.com, b.mx.giacfortunecookies.com / 172.16.1.5, 172.16.1.6
- Hardware: Dell PowerEdge 650 (2.4Ghz Pentium 4, 512MB ECC 266MHz DDR SDRAM, 40Gb IDE 7,200 RPM Hard Drive and 1 Intel Pro+ Ethernet card
- Software: Red Hat GNU/Linux 9.0 (updated on 06.28.03 12:00 EST), daemontools 0.76, ucspi-tcp 0.88, qmail 1.03 + patches, qmai-scanner 1.16, McAfee ViruSCAN for GNU/Linux 4.24, Red Hat Network agent 1.0.2, NET-SNMP 5.0.8 and Veritas NetBackup agent
- Function: External mail servers. These servers are responsible for all incoming and outgoing e-mails of giacfortunecookies.com
- Placement: DMZ segment

Note: Both servers are placed at the DMZ segment and receives all incoming mail from the Intenet for giacfortunecookies.com domain. It also relay e-mail for the internal mail server to the Internet. Before forwarding all incoming messages to the internal server, an anti-virus is used to inspect all messages, looking for a malicious code. If a virus is encoutered, a warning message will be sent to the user and postmaster.

application/service	protocol/port	accepts packets from	destination
SNMP (172.16.1.5 and 172.16.1.6)	listening on 161/UDP	tapioca (src port >1023/UDP)	n/a
SYSLOG (172.16.1.5 and 172.16.1.6)	uses src port >1023/UDP	n/a	514/UDP on goiaba
DNS client	uses src port 53/UDP or >1023/UDP	n/a	53/UDP on a.ns (172.16.1.2) or b.ns (172.16.1.4)
NTP	uses src port 123/UDP	n/a	123/UDP on tick
Red Hat Network agent (172.16.1.5 and 172.16.1.6)	uses src port >1023/TCP	n/a	443/TCP on xmlrpc.rhn.redhat.com
qmail SMTP server (172.16.1.5 and 172.16.1.6)	listening on 25/TCP	SMTP servers outside GIAC's LAN or int.mx (src port >1023/TCP)	n/a
qmail SMTP client (172.16.1.5 and 172.16.1.6)	uses src port >1023/TCP	n/a	25/TCP on int.mx (192.168.2.4) or SMTP servers outside GIAC's LAN
Veritas NetBackup agent	uses src port >1023/TCP	n/a	6101/TCP on cocada
Veritas NetBackup agent	listening on 8192-3/TCP	cocada (src port >1023/TCP)	n/a
OpenSSH	listening on 22/TCP	192.168.1.0/24 (src port >1023/TCP)	n/a

Table 10 - Services on a.mx.giacfortunecookies.com and b.mx.giacfortunecookies.com

1.8.10 - Internal Simple Mail Transport Agent (SMTP) server

- Name(s) / IP address(es): int.mx.giacfortunecookies.com / 192.168.1.4
- Hardware: Dell PowerEdge 650 (2.4Ghz Pentium 4, 512MB ECC 266MHz DDR SDRAM, 40Gb IDE 7,200 RPM Hard Drive and 1 Intel Pro+ Ethernet card
- Software: Red Hat GNU/Linux 9.0 (updated on 06.28.03 12:00 EST), daemontools 0.76, ucspi-tcp 0.88, qmail 1.03 + patches, checkpassword 0.90, Red Hat Network agent 1.0.2, NET-SNMP 5.0.8, WU-imap 2002d and Veritas NetBackup agent
- Function: Internal mail server. Responsible for all user accounts of GIAC Enterprises.

- Placement: Management-Services segment

Note: This servers receives all incoming messages from both external mail servers. Relay all employee's email, sending all messages to the external mail servers. SMTP authentication is used, so when a user needs to send e-mail, it must authenticate first. Users can get their messages with Mozilla client, using imap. SMTP authentication is provided by a patched qmail.

application/service	protocol/port	accepts packets from	destination
SNMP	listening on 161/UDP	tapioca (src port >1023/UDP)	n/a
SYSLOG	uses src port >1023/UDP	n/a	514/UDP on goiaba
DNS client	uses src port 53/UDP or >1023/UDP	n/a	53/UDP on graviola (192.168.2.7
NTP	uses src port 123/UDP	n/a	123/UDP on tick
Red Hat Network agent	uses src port >1023/TCP	n/a	443/TCP on xmlrpc.rhn.redhat.com
qmail SMTP server	listening on 25/TCP	a.mx (172.16.1.5), b.mx (172.16.1.6), 192.168.1.0/24 and 192.168.2.0/24	n/a
qmail SMTP client	uses src port >1023/TCP	n/a	a.mx (172.16.1.5) or b.mx (172.16.1.6)
Veritas NetBackup agent	uses src port >1023/TCP	n/a	6101/TCP on cocada
Veritas NetBackup agent	listening on 8192-3/TCP	cocada (src port >1023/TCP)	n/a
OpenSSH	listening on 22/TCP	192.168.1.0/24 (src port >1023/TCP)	n/a

Table 11 - Services on int.mx.giacfortunecookies.com

1.8.11 - External web cluster servers

- Name(s) / IP address(es): www.giacfortunecookies.com, www-1.giacfortunecookies.com, www-2.giacfortunecookies.com and www-3.giacfortunecookies.com / 172.16.1.33-36

- Hardware: Dell PowerEdge 2650 (dual Xeon 3.06Ghz w/ Hyper-Threading, 1GB DDR SDRAM (ChipKill), 36GB Ultra3 (Ultra160) SCSI drive and 1 Intel Pro/1000XT Ethernet card

- Software:Red Hat GNU/Linux 9.0 (updated on 06.28.03 12:00 EST), Red Hat Network agent 1.0.2, Apache 2.0.46, Apache Tomcat 4.1.24, NET-SNMP 5.0.8 and Veritas NetBackup agent

- Function: Web services

- Placement: Web Cluster segment

Note: Web servers uses clustering technology to provide redundancy and load balancing. The HTTP protocol is used when non sensitive information need to be sent or received, and the HTTPS protocol is used when sensitive information is in transit, which includes all ordering transactions, suppliers and partners activity. Java Server Pages (JSP) technology will be used, and all web clustered servers communicates with the application servers (using Enterprise Java Beans), which are placed on the Database-Applications segment. All communications between the web servers and the application servers are secured using SSL. This imposes an extra load on web servers, but also improves security.

application/service	protocol/port	accepts packets from	destination
SNMP	listening on 161/UDP	tapioca (src port >1023/UDP)	n/a
SYSLOG	uses src port >1023/UDP	n/a	514/UDP on goiaba
DNS client	uses src port 53/UDP or >1023/UDP	n/a	53/UDP on a.ns (172.16.1.2) or b.ns (172.16.1.4)
NTP	uses src port 123/UDP	n/a	123/UDP on tick
Red Hat Network agent	uses src port >1023/TCP	n/a	443/TCP on xmlrpc.rhn.redhat.com
HTTP server	listening on 80,443/TCP	Internet and GIAC's LAN (src port >1023/TCP)	n/a
JBOSS client	uses src port >1023/TCP	n/a	8443/TCP on application servers app1 and app2
Veritas NetBackup agent	uses src port >1023/TCP	n/a	6101/TCP on cocada
Veritas NetBackup agent	listening on 8192-3/TCP	cocada (src port >1023/TCP)	n/a
OpenSSH	listening on 22/TCP	192.168.1.0/24 (src port >1023/TCP)	n/a

Table 12 - Services on www,www-1,www-2,www-3.giacfortunecookies.com

1.8.12 - Intranet server

- Name(s) / IP address(es): intranet.giacfortunecookies.com / 192.168.1.5

- Hardware: Dell PowerEdge 2650 (dual Xeon 3.06Ghz w/ Hyper-Threading, 1GB DDR SDRAM (ChipKill), 36GB Ultra3 (Ultra160) SCSI drive and 1 Intel Pro/1000XT Ethernet card

- Software:Red Hat GNU/Linux 9.0 (updated on 06.28.03 12:00 EST), Red Hat Network agent 1.0.2, Apache 2.0.46, Apache Tomcat 4.1.24, NET-SNMP 5.0.8

- Function: Intranet server

- Placement: Management-Services segment

Note: This server is basically the same as a external web server, but instead of serving pages for selling fortunes, it is used for intranet services, using the same technology of Java Server Pages (JSP) accessing Enterprise Java Beans (EJB) applications at Database-Applications segment.

application/service	protocol/port	accepts packets from	destination
SNMP	listening on 161/UDP	tapioca (src port >1023/UDP)	n/a
SYSLOG	uses src port >1023/UDP	n/a	514/UDP on goiaba
DNS client	uses src port 53/UDP or >1023/UDP	n/a	53/UDP on graviola
NTP	uses src port 123/UDP	n/a	123/UDP on tick
Red Hat Network agent	uses src port >1023/TCP	n/a	443/TCP on xmlrpc.rhn.redhat.com
HTTP server	listening on 80,443/TCP	192.168.2.0/24 and 192.168.4.0/24 (uses src port >1023/TCP)	n/a
HTTP client	uses src port >1023/TCP	n/a	80/TCP on application servers (192.168.3.0/24)
Veritas NetBackup agent	uses src port >1023/TCP	n/a	6101/TCP on cocada
Veritas NetBackup agent	listening on 8192-3/TCP	cocada (src port >1023/TCP)	n/a
OpenSSH	listening on 22/TCP	192.168.1.0/24 (src port >1023/TCP)	n/a

Table 13 - Services on intranet.giacfortunecookies.com

1.8.13 - Network Time Protocol (NTP) server

- Name(s) / IP address(es): tick.giacfortunecookies.com / 192.168.1.1
- Hardware: Dell PowerEdge 650 (2.4Ghz Pentium 4, 128MB ECC 266MHz DDR SDRAM, 40Gb IDE 7,200 RPM Hard Drive and 1 Intel Pro+ Ethernet card
- Software: Red Hat GNU/Linux 9.0 (updated on 06.28.03 12:00 EST), Red Hat Network agent 1.0.2, ntpd 4.1.1, NET-SNMP 5.0.8
- Function: Time synchronization service for all servers, devices and workstations of GIAC Enterprises's LAN
- Placement: Management-Services segment

Note: This server will provide time synchronization service for the GIAC Enterprises's LAN. It will synchronize itself with 3 public Stratum-1 or Stratum-2 servers, in order to provide a reliable service. It is very important to have all servers, devices and workstations with the correct time, in order to know exactly at what time an event occured. It is also important when sending logs to a Incident Response Team.

application/service	protocol/port	accepts packets from	destination
SNMP	listening on 161/UDP	tapioca (src port >1023/UDP)	n/a
SYSLOG	uses src port >1023/UDP	n/a	514/UDP on goiaba
DNS client	uses src port 53/UDP or >1023/UDP	n/a	53/UDP on graviola
NTP	uses src port 123/UDP	n/a	123/UDP on tick
Red Hat Network agent	uses src port >1023/TCP	n/a	443/TCP on xmlrpc.rhn.redhat.com
Veritas NetBackup agent	uses src port >1023/TCP	n/a	6101/TCP on cocada
Veritas NetBackup agent	listening on 8192-3/TCP	cocada (src port >1023/TCP)	n/a
OpenSSH	listening on 22/TCP	192.168.1.0/24 (src port >1023/TCP)	n/a

Table 14 - Services on tick.giacfortunecookies.com

1.8.14 - Database servers

- Name(s) / IP address(es): db1.giacfortunecookies.com and db2.giacfortunecookies.com/ 192.168.3.4-5
- Hardware: Dell PowerEdge 2650 (dual Xeon 3.06Ghz w/ Hyper-Threading, 2GB DDR SDRAM (ChipKill), 73Ultra3 (Ultra160) SCSI drive and 1 Intel Pro/1000XT Ethernet card
- Software: Red Hat GNU/Linux 9.0 (updated on 06.28.03 12:00 EST), Red Hat Network agent 1.0.2, Oracle 9i, NET-SNMP 5.0.8 and Veritas NetBackup agent
- Function: Backend database server
- Placement: Database-Application segment

Note: Oracle 9i for GNU/Linux is used as a backend database service for the Enterprise Java Beans Applications. It uses its own clustering technology.

application/service	protocol/port	accepts packets from	destination
SNMP	listening on 161/UDP	tapioca (src port >1023/UDP)	n/a
SYSLOG	uses src port >1023/UDP	n/a	514/UDP on goiaba

DNS client	uses src port 53/UDP or >1023/UDP	n/a	53/UDP on graviola
NTP	uses src port 123/UDP	n/a	123/UDP on tick
Red Hat Network agent	uses src port >1023/TCP	n/a	443/TCP on xmlrpc.rhn.redhat.com
Oracle 9i	listening on 1521/TCP	app1 and app2 (src port >1023/TCP)	n/a
Veritas NetBackup agent	uses src port >1023/TCP	n/a	6101/TCP on cocada
Veritas NetBackup agent	listening on 8192-3/TCP	cocada (src port >1023/TCP)	n/a
OpenSSH	listening on 22/TCP	192.168.1.0/24 and 192.168.3.0/24 (src port >1023/TCP)	n/a

Table 15 - Services on db1.giacfortunecookies.com and db2.giacfortunecookies.com

1.8.15 - Application cluster servers

- Name(s) / IP address(es): app1.giacfortunecookies.com and app2.giacfortunecookies.com, / 192.168.3.8-9

- Hardware: Dell PowerEdge 2650 (dual Xeon 3.06Ghz w/ Hyper-Threading, 2GB DDR SDRAM (ChipKill), 73Ultra3 (Ultra160) SCSI drive and 1 Intel Pro/1000XT Ethernet card

- Software: Red Hat GNU/Linux 9.0 (updated on 06.28.03 12:00 EST), Red Hat Network agent 1.0.2, Jboss 3.2, NET-SNMP 5.0.8 and Veritas NetBackup agent

- Function: Enterprise JavaBeans application servers

- Placement: Database-Application segment

Note: JBoss is a Open Source Enterprise JavaBeans Application server implemented in Java. JBoss supports EJB container and JMX infrastructure. It is used to serve Java applications that will be accessed by JSP pages via Apache TomCat.

application/service	protocol/port	accepts packets from	destination
SNMP	listening on 161/UDP	tapioca (src port >1023/UDP)	n/a
SYSLOG	uses src port >1023/UDP	n/a	514/UDP on goiaba
DNS client	uses src port 53/UDP or >1023/UDP	n/a	53/UDP on graviola
NTP	uses src port 123/UDP	n/a	123/UDP on tick

Red Hat Network agent	uses src port >1023/TCP	n/a	443/TCP on xmlrpc.rhn.redhat.com
Jboss server	listening on 80/TCP	www, www-1, www-2, www-3 and intranet (src port >1023/TCP)	n/a
Jboss client	uses port >1023/TCP	n/a	1521/TCP on db1 or db2 (src port >1023/TCP)
Veritas NetBackup agent	uses src port >1023/TCP	n/a	6101/TCP on cocada
Veritas NetBackup agent	listening on 8192-3/TCP	cocada (src port >1023/TCP)	n/a
OpenSSH	listening on 22/TCP	192.168.1.0/24 and 192.168.3.0/24 (src port >1023/TCP)	n/a

Table 16 - Services on app1.giacfortunecookies.com and app2.giacfortunecookies.com

1.8.16 - Proxy Server

- Name(s) / IP address(es): doritos.giacfortunecookies.com / 192.168.1.6
- Hardware: Dell PowerEdge 2650 (dual Xeon 3.06Ghz w/ Hyper-Threading, 1GB DDR SDRAM (ChipKill), 36GB Ultra3 (Ultra160) SCSI drive and 1 Intel Pro/1000XT Ethernet card
- Software: Red Hat GNU/Linux 9.0 (updated on 06.28.03 12:00 EST), Red Hat Network agent 1.0.2, Squid 2.5-Stable 3 and SquidGuard 1.2.0
- Function: Provide web services for all GIAC Enterprises Employees and control certain types of content.
- Placement: Management-Services segment

Note: The proxy server enable all employees to browse the Internet and save bandwidth. All employees must authenticate first, in order to have access to browse the web. It is prohibited downloading files with extensions like .exe, .bin. .scr, .pif, among others. Squid is used as the proxy server, and will listen on port 3128.

application/service	protocol/port	accepts packets from	destination
SNMP	listening on 161/UDP	tapioca (src port >1023/UDP)	n/a
SYSLOG	uses src port >1023/UDP	n/a	514/UDP on goiaba
DNS client	uses src port 53/UDP or >1023/UDP	n/a	53/UDP on graviola
NTP	uses src port 123/UDP	n/a	123/UDP on tick
Red Hat Network agent	uses src port >1023/TCP	n/a	443/TCP on xmlrpc.rhn.redhat.com

Squid server	listening on 3128/TCP	192.168.2.0/24 and 192.168.4.0/24 (src port >1023/TCP)	n/a
Squid client	uses src port >1023/TCP	n/a	21, 80, 443/TCP on servers outside GIAC's LAN
Squid LDAP auth	uses src port >1023/TCP	n/a	389/TCP on quindim
Veritas NetBackup agent	uses src port >1023/TCP	n/a	6101/TCP on cocada
Veritas NetBackup agent	listening on 8192-3/TCP	cocada (src port >1023/TCP)	n/a
OpenSSH	listening on 22/TCP	192.168.1.0/24 (src port >1023/TCP)	n/a

Table 17 - Services on doritos.giacfortunecookies.com

1.8.17 - File and Printing Services

- Name(s) / IP address(es): quindim.giacfortunecookies.com / 192.168.1.2
- Hardware: Dell PowerEdge 2650 (dual Xeon 3.06Ghz w/ Hyper-Threading, 1GB DDR SDRAM (ChipKill), 36GB Ultra3 (Ultra160) SCSI drive and 1 Intel Pro/1000XT Ethernet card
- Software: Red Hat GNU/Linux 9.0 (updated on 06.28.03 12:00 EST), Red Hat Network agent 1.0.2, Samba 2.2.8a
- Function: File sharing and print services
- Placement: Management-Services segment

Note: Samba is being used to provide file sharing and printing services. All users must authenticate, in order to have access to their own files or printing documents. LDAP is used to provide a backed databased used by Samba for user authentication.

application/service	protocol/port	accepts packets from	destination
SNMP	listening on 161/UDP	tapioca (src port >1023/UDP)	n/a
SYSLOG	uses src port >1023/UDP	n/a	514/UDP on goiaba
DNS client	uses src port 53/UDP or >1023/UDP	n/a	53/UDP on graviola
NTP	uses src port 123/UDP	n/a	123/UDP on tick
Red Hat Network agent	uses src port >1023/TCP	n/a	443/TCP on xmlrpc.rhn.redhat.com
Samba	listening on 137/UDP,	192.168.2.0/24 (uses src port	n/a

	138/UDP and 139/TCP	>1023/UDP,TCP)	
OpenLDAP server	listening on 389/TCP	doritos and localhost (src port >1023/TCP)	n/a
Veritas NetBackup agent	uses src port >1023/TCP	n/a	6101/TCP on cocada
Veritas NetBackup agent	listening on 8192-3/TCP	cocada (src port >1023/TCP)	n/a
OpenSSH	listening on 22/TCP	192.168.1.0/24 (src port >1023/TCP)	n/a

Table 18 - Services on quindim.giacfortunecookies.com

1.8.18 - Backup Server

- Name(s) / IP address(es): cocada.giacfortunecookies.com / 192.168.1.9
- Hardware: Dell PowerEdge 2650 (dual Xeon 3.06Ghz w/ Hyper-Threading, 512MB DDR SDRAM (ChipKill), 73GB Ultra3 (Ultra160) SCSI drive and 1 Intel Pro/1000XT Ethernet card
- Software: Red Hat GNU/Linux 9.0 (updated on 06.28.03 12:00 EST), Red Hat Network agent 1.0.2, Veritas NetBackup 4.5
- Function: Centralized backup station
- Placement: Management-Services segment

Note: Veritas NetBackup Professional for GNU/Linux will be used to mananage the backup procedure for all servers. Each server will run a small client that will communicate with the backup server on a scheduled time. Veritas NetBackup is a well known backup software and reliable.

application/service	protocol/port	accepts packets from	destination
SNMP	listening on 161/UDP	tapioca (src port >1023/UDP)	n/a
SYSLOG	uses src port >1023/UDP	n/a	514/UDP on goiaba
DNS client	uses src port 53/UDP or >1023/UDP	n/a	53/UDP on graviola
NTP	uses src port 123/UDP	n/a	123/UDP on tick
Red Hat Network agent	uses src port >1023/TCP	n/a	443/TCP on xmlrpc.rhn.redhat.com
Veritas NetBackup server	listening on 6101/TCP	172.16.1.0/27, 172.16.1.32/27, 192.168.2.0/24, 192.168.3.0/24, 192.168.4.0/24 and 192.168.255.0/24 (src	n/a

		port >1023/TCP)	
Veritas NetBackup client	uses src port >1023/TCP	n/a	8192-3/TCP on 172.16.1.0/27, 172.16.1.32/27, 192.168.2.0/24, 192.168.3.0/24
OpenSSH	listening on 22/TCP	192.168.1.0/24 (src port >1023/TCP)	n/a

Table 19 - Services on cocada.giacfortunecookies.com

1.8.19 - SNMP monitoring station

- Name(s) / IP address(es): tapioca.giacfortunecookies.com / 192.168.1.10
- Hardware: Dell PowerEdge 650 (2.4Ghz Pentium 4, 256MB ECC 266MHz DDR SDRAM, 40Gb IDE 7,200 RPM Hard Drive and 1 Intel Pro+ Ethernet card
- Software: Red Hat GNU/Linux 9.0 (updated on 06.28.03 12:00 EST), Red Hat Network agent 1.0.2, MRTG 2.10.0pre3, Apache 2.0.46, Veritas NetBackup 4.5
- Function: SNMP monitoring/management station
- Placement: Management-Services segment

Note: SNMP is used to monitor the health of every device or server of GIAC Enterprises LAN. Information of interface, memory and CPU utilization are de basic information collected. Other scripts are used but it is out of scope of this document discuss how these scripts work.

application/service	protocol/port	accepts packets from	destination
SNMP	uses src port >1023/UDP	n/a	172.16.1.0/27, 172.16.1.32/27, 192.168.2.0/24 and 192.168.3.0/24
SYSLOG	uses src port >1023/UDP	n/a	514/UDP on goiaba
DNS client	uses src port 53/UDP or >1023/UDP	n/a	53/UDP on graviola
NTP	uses src port 123/UDP	n/a	123/UDP on tick
Red Hat Network agent	uses src port >1023/TCP	n/a	443/TCP on xmlrpc.rhn.redhat.com
Veritas NetBackup agent	uses src port >1023/TCP	n/a	6101/TCP on cocada
Veritas NetBackup agent	listening on 8192-3/TCP	cocada (src port >1023/TCP)	n/a
OpenSSH	listening on	192.168.1.0/24	n/a

	22/TCP	(src port >1023/TCP)	

Table 20 - Services on tapioca.giacfortunecookies.com

1.8.20 - IDS managment workstation

- Name(s) / IP address(es): cafe.giacfortunecookies.com / 192.168.255.10
- Hardware: Dell PowerEdge 650 (2.4Ghz Pentium 4, 512MB ECC 266MHz DDR SDRAM, 40Gb IDE 7,200 RPM Hard Drive and 1 Intel Pro+ Ethernet card
- Software: Red Hat GNU/Linux 9.0 (updated on 06.28.03 12:00 EST), Red Hat Network agent 1.0.2, Apache 2.0.46, PHP 4.2.2, MySQL 4.0.13, Acid 0.9.6b23, Veritas NetBackup 4.5
- Function: Centralize all IDS logs for further analysis with Acid
- Placement: IDS segment

Note: This server is used to collect and manage all attack informatio detected by the IDS sensors placed on the different segments of the LAN. Analysis Console for Intrusion Databases (ACID) is used to search the database for consolidation of events detected. It is possible to find alers matching on alert meta information (signature, detection time, etc.) as well as the underlying network evidence (source/destination address, ports, payload or flags). It also does charts and statistics generation based on time, sensor, signature, protocol, IP address, TCP/UDP ports, or classification. This is a powerful tool that will be used to ease the administration of Intrusion Detection Systems.

application/service	protocol/port	accepts packets from	destination
SNMP	listening on 161/UDP	tapioca (src port >1023/UDP)	n/a
SYSLOG	uses src port >1023/UDP	n/a	514/UDP on goiaba
DNS client	uses src port 53/UDP or >1023/UDP	n/a	53/UDP on graviola
NTP	uses src port 123/UDP	n/a	123/UDP on tick
Red Hat Network agent	uses src port >1023/TCP	n/a	443/TCP on xmlrpc.rhn.redhat.com
MySQL server	listening on 3306/TCP	s1, s2, s3, s4 and s5 (uses src port >1023/TCP)	n/a
OpenSSH	listening on 22/TCP	192.168.255.0/24 (src port >1023/TCP)	n/a

Table 21 - Services on cafe.giacfortunecookies.com

1.8.21 - Users Workstations

- Name(s) / IP address(es): ws-[1-32].giacfortunecookies.com / 192.168.2.2-32
- Hardware: Dell Precision 360 (2.4Ghz Pentium 4, 256MB ECC 266Mhz DDR SDRAM, 40GB IDE 7,200 RPM hard Drive and 1 Intel Pro+ Ethernet card
- Software: Microsoft Windows XP, Microsoft Office XP, Netscape 7.1, Norton anti-virus 2003, Edudora 5.2
- Function: Employee workstation, used on day-to-day work.
- Placement: Internal users segment

Note:

application/service	protocol/port	accepts packets from	destination
active FTP client	uses src port >1023/TCP	n/a	3128/TCP on doritos
SMTP client	uses src port >1023/TCP	n/a	25/TCP on int.mx
DNS client	uses src port >1023/UDP	n/a	53/UDP on graviola
DHCP client	0.0.0.0 src port 68/UDP	n/a	67/UDP on 255.255.255.255
SMB client	uses src port >1023/TCP,UDP	n/a	137/UDP, 138/UDP and 139/TCP on quindim
IMAP client	uses src port >1023/TCP	n/a	143/TCP on int.mx
HTTP	uses src port >1023/TCP	n/a	3128/TCP on doritos
HTTPS	uses src port >1023/TCP	n/a	3128/TCP on doritos

Table 22 - Services on ws-[1-32].giacfortunecookies.com

1.8.22 - DHCP server

- Name(s) / IP address(es): batata.giacfortunecookies.com / 192.168.4.1
- Hardware: Dell PowerEdge 650 (2.4Ghz Pentium 4, 128MB ECC 266MHz DDR SDRAM, 40Gb IDE 7,200 RPM Hard Drive and 1 Intel Pro+ Ethernet card
- Software: Red Hat GNU/Linux 9.0 (updated on 06.28.03 12:00 EST), Red Hat Network agent 1.0.2, Apache 2.0.46, PHP 4.2.2, MySQL 4.0.13, Acid 0.9.6b23, Veritas NetBackup 4.5
- Function: DHCP server
- Placement: Internal users segment

Note: This server is used for automatic configuration of IP address, network mask, default gateway and DNS server. It is logged each IP lease for further analysis.

application/service	protocol/port	accepts packets from	destination
SNMP	listening on 161/UDP	tapioca (src port >1023/UDP)	n/a
SYSLOG	uses src port >1023/UDP	n/a	514/UDP on goiaba
DNS client	uses src port 53/UDP or >1023/UDP	n/a	53/UDP on graviola
NTP	uses src port 123/UDP	n/a	123/UDP on tick
DHCP server	listening broadcast on port 67/UDP	255.255.255.255 (src port 68)	n/a
DHCP server	a.b.c.d src port 67/UDP	n/a	68/UDP on 255.255.255.255
Red Hat Network agent	uses src port >1023/TCP	n/a	443/TCP on xmlrpc.rhn.redhat.com
Veritas NetBackup agent	uses src port >1023/TCP	n/a	6101/TCP on cocada
Veritas NetBackup agent	listening on 8192-3/TCP	cocada (src port >1023/TCP)	n/a
OpenSSH	listening on 22/TCP	192.168.2.0/24 and 192.168.255.0/24 (src port >1023/TCP)	n/a

Table 23 - Services on batata.giacfortunecookies.com

1.8.23 - Switches

- Name(s) / IP address(es): sw-[1-5].giacfortunecookies.com /
- Hardware: Cisco Catalyst 3550 24 FX
- Software: Standard Multilayer Software Image (SMI)
- Function: Traffic distribution
- Placement: all segments

Note: 5 Catalyst swiches are used for traffic distribution on GIAC Enterprises network. Each switch will be configured to have one port mirroring, which is connected with the IDS.

application/service	protocol/port	accepts packets from	destination
SNMP	listening on 161/UDP	tapioca (src port >1023/UDP)	n/a
SYSLOG	uses src port >1023/UDP	n/a	514/UDP on goiaba
DNS client	uses src port 53/UDP or >1023/UDP	n/a	53/UDP on graviola

Assignment 2 - Security Policy and Tutorial

2.1 - Border Router Security Policy

The border router is the very first line of defense of GIAC Enterprises, and does some basic filtering in order to protect itself and devices behind its domain, from spoofed packets, ICMP, UDP and TCP floods using Cisco's Committed Access Rate (CAR) feature. All logs are sent to the loghost, for log centralization and further analysis. SNMP and NetFlow are used for monitoring the the health and traffic characterization, respectively.

2.1.1 - Global configuration

`hostname abacaxi`	Assign the router's name.
`service nagle`	Enable the Nagle congestion control algorithm defined on RFC 896.
`service tcp-keepalives-in`	Generate keepalive packets on idle incoming network connections.
`service tcp-keepalives-out`	Generate keepalive packets on idle outgoing network connections.
`service timestamps debug datetime msec show-timezone localtime`	Define the timestamp on debug messages.
`service timestamps log datetime msec show-timezone localtime`	Define the timestamp on log messages.
`service password-encryption`	Encrypts passwords on configuration file when written on memory. This command does not provide high security and it necessary to use other security methods to improve security.
`enable secret`	Use MD5 algorithm to hash 'enable' password.
`logging buffered 8192 warnings`	Limit the internal log buffer to 8192 bytes and defines log severity.
`logging console critical`	Limit message logs to console, based on severity.
`logging 192.168.2.4`	Define the syslog server host (loghost).
`no service dhcp`	Disable the DCHP server support on router.
`no service pad`	Disable Packet Assembler/Disassembler (PAD) commands and connections support (X.25 support).
`no service finger`	Disable finger support.
`no service tcp-small-servers`	Disable Echo, Discard and Chargen support.
`no service udp-small-servers`	Disable Echo, Discard and Chargen support.

`ip classless`	Enable packet forwarding to the best supernet available.
`ip subnet-zero`	Enable usage and routing to subnet 0.
`ip cef`	Enables Cisco Express Forwarding (CEF). It optimizes network/switching performance of the router. Other IOS features need CEF to be enabled (eg. NetFlow).
`ip tcp intercept mode intercept`	Change interception to active mode. TCP Intercept protects TCP servers from TCP SYN flooding attacks.
`ip tcp intercept list 102`	Define the access-list used on TCP intercept duties.
`ip tcp intercept connection-timeout 60`	
`ip tcp intercept watch-timeout 10`	
`ip tcp intercept one-minute low 2000`	
`ip tcp intercept one-minute high 6000`	
`no ip source-route`	Disable handling IP datagrams with source routing header options.
`no ip finger`	Disable the finger support.
`no ip http server`	Disable http server support.
`no ip identd`	Disable the identd support.
`no ip bootp server`	Disable the Bootp support.
`no ip domain-lookup`	Disable DNS lookup support.
`no ip rcmd rsh-enable`	Disable Remote Shell (rsh) commands support.
`no ip rcmd rcp-enable`	Disable Remote Copy (rcp) command support.
`no cdp run`	Disable the Cisco Discovery Protocol (CDP).
`clock timezone GMT -3`	Define the correct timezone for São Paulo state.
`clock summer-time GMT -2 date Nov 3 2002 0:00 Feb 16 2003 0:00`	Define the Daylight Saving Time for 2003/2004 period (based on local government decreet).
`ntp server 172.16.1.32`	Define the Network Time Protocol (NTP) server to adjust the internal clock.
`snmp-server community abacaxi-giac RO 100`	Define the SNMP community (read-only) and access-list used to control access to the SNMP service.
`aaa new-model`	Enable the Authentication, Authorization and Accounting (AAA) access control model.
`aaa authentication login default local`	Sets AAA authentication at login. If the default list is not set, the local user database is used instead.

aaa authentication enable default enable	Enable AAA to determine if a user can access the privileged command level.
username <username> password <password>	Sets the <username> password.
logging trap debugging	Defines the log severity for all events sent to loghost.
logging facility local7	Defines the SYSLOG facility used to tag log messages.
logging source-interface loopback0	
logging 192.168.2.3	Sets the loghost to be used.
ip flow-export source loopback0	Sets the interface used for src IP address. It does not mean flow information of the loopback0 interface.
ip flow-export destination 192.168.2.10	Sets the detination of NetFlow data.
ip flow-export version 5 origin-as	Sets the NetFlow version to be used.

2.1.2 - Loopback0 interface configuration

interface loopback0	
ip address 10.10.10.10 255.255.255.255	Defines the IP address used on the interface.
no ip redirects	Disable sending Internet Control Message Protocol (ICMP) redirect messages on this interface.
no ip unreachables	Disable sending Internet Control Message Protocol (ICMP) unreachable messages on this interface.
no ip proxy-arp	Disable proxy Address Resolution Protocol (ARP) on this interface.

2.1.3 - Null0 interface configuration

This interface is used to blackhole routes.

interface null0	
no ip unreachables	Disable sending Internet Control Message Protocol (ICMP) unreachable messages on this interface.

2.1.4 Ethernet0/0 interface configuration

This is the interface facing external connection with the ISP. Some services are disabled to avoid security problems. The access-list 1001 is used in order to deny the RFC 1918 blocks and all IANA IPv4 unallocated blocks. It is necessary monitor changes in netblock allocations, periodically visiting the IANA website (http://www.iana.org/assignments/ipv4-address-space).

```
interface Ethernet0/0
```

`ip address a.b.c.254 255.255.255.0`	Define the IP address used for the external interface.
`ip verify unicast reverse-path`	Configure the router to make sure that the source address of a IP datagram appears in the routing table and matches the interface on which the packet was received. This feature is very useful on detecting Denial of Service (DoS) attacks. Receiving malformed packets are a good indication of an attack. It Can only be used if the path is symetric.
`ip access-group 1001 in`	Defines the access-list used for this interface.
`no ip redirects`	Disable sending ICMP redirect messages on this interface.
`no ip unreachables`	Disable sending ICMP unreachable messages on this interface.
`no ip directed-broadcast`	Configure the router to drop all IP packets to broadcast addresses.
`no ip proxy-arp`	Disable proxy Address Resolution Protocol (ARP) on this in terface.
`no ip mask-reply`	Configure the router to not answer ICMP messages mask requests with ICMP mask reply messages.
`ip accounting access-violations`	Enables IP accounting and look for IP traffic that fails IP access lists.
`ip route-cache flow`	Enable NetFlow accounting.

2.1.5 Ethernet1/0 interface configuration

`interface Ethernet0/1`	
`ip address 172.16.1.249 255.255.255.240`	Define the IP address used for the internal interface.
`ip verify unicast reverse-path`	Make sure that the source address of a IP datagram appears in the routing table and matches the interface on wich the packet was received.
`ip access-group 1002 in`	Defines the access-list used for this interface.
`no ip redirects`	Disable sending ICMP redirect messages on this interface.
`no ip unreachables`	Disable sending ICMP unreachable messages on this interface.
`no ip directed-broadcast`	Configure the router to drop all IP packets to broadcast addresses
`no ip proxy-arp`	Disable proxy Address Resolution Protocol (ARP) on this interface.
`no ip mask-reply`	Configure the router to not answer ICMP messages mask requests with ICMP mask reply messages.

2.1.6 Ethernet0/2 interface configuration

This interface is disabled.

```
interface Ethernet 0/2
no description
no ip address
shutdown
```

2.1.7 Ethernet0/3 interface configuration

This interface is disabled.

```
interface Ethernet 0/3
no description
no ip address
shutdown
```

2.1.8 - Routes

Here is defined the default route (it could be a routing protocol instead), static routes to reach the internal network and blackhole bogus routes to mitigate spoofing problems. This includes the recomendation of RFC 1918 and all reserved routes listed by IANA.

```
ip route 0.0.0.0 0.0.0.0 a.b.c.1
ip route 172.16.1.0 255.255.255.0 172.16.1.254
ip route 192.168.0.0 255.255.0.0 172.16.1.254

ip route 1.0.0.0 255.0.0.0 null0
ip route 2.0.0.0 255.0.0.0 null0
ip route 5.0.0.0 255.0.0.0 null0
ip route 7.0.0.0 255.0.0.0 null0
ip route 10.0.0.0 255.0.0.0 null0
ip route 23.0.0.0 255.0.0.0 null0
ip route 27.0.0.0 255.0.0.0 null0
ip route 31.0.0.0 255.0.0.0 null0
ip route 36.0.0.0 255.0.0.0 null0
ip route 37.0.0.0 255.0.0.0 null0
ip route 39.0.0.0 255.0.0.0 null0
ip route 49.0.0.0 255.0.0.0 null0
ip route 50.0.0.0 255.0.0.0 null0
ip route 58.0.0.0 255.0.0.0 null0
ip route 59.0.0.0 255.0.0.0 null0
ip route 70.0.0.0 255.0.0.0 null0
ip route 71.0.0.0 255.0.0.0 null0
ip route 72.0.0.0 255.0.0.0 null0
ip route 73.0.0.0 255.0.0.0 null0
ip route 74.0.0.0 255.0.0.0 null0
ip route 75.0.0.0 255.0.0.0 null0
ip route 76.0.0.0 255.0.0.0 null0
ip route 77.0.0.0 255.0.0.0 null0
ip route 78.0.0.0 255.0.0.0 null0
ip route 79.0.0.0 255.0.0.0 null0
ip route 83.0.0.0 255.0.0.0 null0
ip route 84.0.0.0 255.0.0.0 null0
ip route 85.0.0.0 255.0.0.0 null0
ip route 86.0.0.0 255.0.0.0 null0
ip route 87.0.0.0 255.0.0.0 null0
ip route 88.0.0.0 255.0.0.0 null0
ip route 89.0.0.0 255.0.0.0 null0
ip route 90.0.0.0 255.0.0.0 null0
ip route 91.0.0.0 255.0.0.0 null0
ip route 92.0.0.0 255.0.0.0 null0
ip route 93.0.0.0 255.0.0.0 null0
ip route 94.0.0.0 255.0.0.0 null0
ip route 95.0.0.0 255.0.0.0 null0
ip route 96.0.0.0 255.0.0.0 null0
ip route 97.0.0.0 255.0.0.0 null0
ip route 98.0.0.0 255.0.0.0 null0
ip route 99.0.0.0 255.0.0.0 null0
ip route 100.0.0.0 255.0.0.0 null0
ip route 101.0.0.0 255.0.0.0 null0
ip route 102.0.0.0 255.0.0.0 null0
ip route 103.0.0.0 255.0.0.0 null0
ip route 104.0.0.0 255.0.0.0 null0
ip route 105.0.0.0 255.0.0.0 null0
ip route 106.0.0.0 255.0.0.0 null0
ip route 107.0.0.0 255.0.0.0 null0
ip route 108.0.0.0 255.0.0.0 null0
ip route 109.0.0.0 255.0.0.0 null0
ip route 110.0.0.0 255.0.0.0 null0
ip route 111.0.0.0 255.0.0.0 null0
ip route 112.0.0.0 255.0.0.0 null0
ip route 113.0.0.0 255.0.0.0 null0
ip route 114.0.0.0 255.0.0.0 null0
ip route 115.0.0.0 255.0.0.0 null0
ip route 116.0.0.0 255.0.0.0 null0
ip route 117.0.0.0 255.0.0.0 null0
ip route 118.0.0.0 255.0.0.0 null0
ip route 119.0.0.0 255.0.0.0 null0
ip route 120.0.0.0 255.0.0.0 null0
ip route 121.0.0.0 255.0.0.0 null0
ip route 122.0.0.0 255.0.0.0 null0
ip route 123.0.0.0 255.0.0.0 null0
ip route 124.0.0.0 255.0.0.0 null0
ip route 125.0.0.0 255.0.0.0 null0
ip route 126.0.0.0 255.0.0.0 null0
ip route 127.0.0.0 255.0.0.0 null0
ip route 173.0.0.0 255.0.0.0 null0
ip route 174.0.0.0 255.0.0.0 null0
ip route 175.0.0.0 255.0.0.0 null0
ip route 176.0.0.0 255.0.0.0 null0
ip route 177.0.0.0 255.0.0.0 null0
ip route 178.0.0.0 255.0.0.0 null0
ip route 179.0.0.0 255.0.0.0 null0
ip route 180.0.0.0 255.0.0.0 null0
ip route 181.0.0.0 255.0.0.0 null0
ip route 182.0.0.0 255.0.0.0 null0
ip route 183.0.0.0 255.0.0.0 null0
ip route 184.0.0.0 255.0.0.0 null0
ip route 185.0.0.0 255.0.0.0 null0
ip route 186.0.0.0 255.0.0.0 null0
ip route 187.0.0.0 255.0.0.0 null0
ip route 189.0.0.0 255.0.0.0 null0
ip route 190.0.0.0 255.0.0.0 null0
!ip route 192.168.0.0 255.255.0.0 null0
ip route 197.0.0.0 255.0.0.0 null0
ip route 223.0.0.0 255.0.0.0 null0
ip route 224.0.0.0 255.0.0.0 null0
```

2.1.9 - Access List 100 - ACL for SNMP

Access list for accessing SNMP on the router, which only accepts packets from 172.16.1.254/32.

```
access-list 100 permit 192.168.1.1
access-list 100 deny any log
```

2.1.10 - Access List 102 - ACL for TCP intercept

Access list used for TCP Intercept. It only perform protection for 172.16.1.0/24.

```
access-list 102 permit 172.16.1.0 0.0.0.255
```

2.1.11 - Access List 104 - ACL for VTY access

Access list for VTY access via Telnet or SSH. It only accepts packets fom the host 172.16.1.254.

```
access-list 104 permit tcp host 172.16.1.254 host 0.0.0.0
range 22 23 log-input
access-list 104 deny ip any any log-inpupt
```

2.1.12 - Access List 1001 - ACL for the internal interface

Access list used on the interface facing connection with the ISP. First, blocks spoofed packets that apparently comming from the internal network then denies all RFC1918 and IANA unallocated netblocks and ICMP fragments. It only accepts traffic for the 172.16.1.0/24 and Multicast netblocks.

```
access-list 1001 deny ip 172.16.1.0 0.0.0.255 any log-input

access-list 1001 deny ip 0.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 1.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 2.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 5.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 7.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 10.0.0.0 0.255.255.255 any log-input
```

```
access-list 1001 deny ip 23.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 27.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 31.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 36.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 37.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 39.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 41.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 42.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 49.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 50.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 58.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 59.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 70.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 71.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 72.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 73.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 74.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 75.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 76.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 77.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 78.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 79.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 83.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 84.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 85.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 86.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 87.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 88.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 89.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 90.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 91.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 92.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 93.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 94.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 95.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 96.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 97.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 98.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 99.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 100.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 101.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 102.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 103.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 104.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 105.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 106.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 107.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 108.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 109.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 110.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 111.0.0.0 0.255.255.255 any log-input
```

```
access-list 1001 deny ip 112.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 113.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 114.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 115.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 116.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 117.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 118.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 119.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 120.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 121.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 122.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 123.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 124.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 125.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 126.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 127.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 173.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 174.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 175.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 176.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 177.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 178.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 179.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 180.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 181.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 182.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 183.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 184.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 185.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 186.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 187.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 189.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 190.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 197.0.0.0 0.255.255.255 any log-input
access-list 1001 deny ip 223.0.0.0 0.255.255.255 any log-input

access-list 1001 deny icmp any any fragments log-input

access-list 1001 permit ip any 172.16.1.0 0.0.0.255
access-list 1001 permit ip any 224.0.0.0 15.255.255.255

access-list 1001 deny ip any any log-input
```

2.1.13 - Access List 1002 - ACL for the internal interface

Access list used on the interface facing the internal network. It permits ICMP echo (inbound ping), echo-reply (ping response), path MTU discovery, time-exceeded (for traceroute) and all traffic from the internal network (172.16.1.0/24), everything else is dropped.

```
access-list 1002 deny icmp any any fragments log-input
access-list 1002 permit icmp any any echo
```

```
access-list 1002 permit icmp any any echo-reply
access-list 1002 permit icmp any any packet-too-big
access-list 1002 permit icmp any any time-exceeded
access-list 1002 deny icmp any any log-input

access-list 1002 permit ip 172.16.1.0 0.0.0.255 any
access-list 1002 deny ip any any log-input
```

2.1.14 - Banner and VTY configuration

A banner can't do much, but warns the intruder that is illegal an unauthorized access to the router.

```
! banner
banner motd %

If you're reading this, it is supposed you're a GIAC
Enterprise authorized employee.
Violators will be prossecuted. Boo! Go away!

%

!
line console 0
exec-timeout 15 0
password
login
!
line vty 0 4
transport input telnet ssh
exec-timeout 15 0
access-class 104
password
login
!
end
```

2.2 - Primary Firewall

The primary firewall is the second layer of defense of GIAC Enterprises. laranja is a Red Hat GNU/Linux with Netfilter/IPTABLES and its main function is to protect itself, and all public visible segments (DMZ and WEBC). All rules of this firewall are derived from the component specification on section 1.8. That means every component that communicates with the DMZ, WEBC and outside GIAC's network has its communication controlled by this firewall.

2.2.1 - rc.firewall firewall initialization script for Netfiler/IPTABLES on laranja

The script below is executed by the initialization script rc.local every time the firewall boots up. All necessary instructions to NETFILTER/IPTABLES is defined in this script.

```
#/bin/sh
#$Id: rc.firewall, v 0.03 18:15 GMT -3 09/03/2003 alexcm Exp $
```

Variables definition. Netfiter/IPTABLES can be automated by shell scripts, that helps a lot the system administrator. It is defined all firewall's Ethernet interfaces, network segments and hosts.

```
# Load all necessary kernel modules used in this firewall
/sbin/modprobe ip_tables
/sbin/modprobe ip_conntrack
/sbin/modprobe ip_conntrack_ftp

IPTABLES="/usr/sbin/iptables"

LOOPBACK_IFACE="lo"       # Loopback interface
PERIM_IFACE="eth0"              # Perimeter interface.
Perimeter is the region between the firewall and border router
DMZ_IFACE="eth1"                # DMZ interface. This
interface is connected to the DMZ
WEBC_IFACE="eth2"               # WEBC interface. This
interface is connected to the Web Cluster segment
INTERM_IFACE="eth3"             # INTERM interface. This
interface is connected to the intermediary area (between
firewalls)

LARANJA="172.16.1.249"                # IP address definition
for the interface facing the perimeter segment
LARANJA_INTERM="172.16.1.246"         # IP address definition
for the interface facing the intermediary segment
LARANJA_WEBC="172.16.1.62"            # IP address definition
for the interface facing the web cluster segment
LARANJA_DMZ="172.16.1.30"       # IP address definition for
the interface facing the DMZ segment

GIAC_NET="172.16.1.0/24"        # GIAC's entire IP address
space
MNGT_NET="192.168.1.0/24"       # Services/Management subnet
USERS_NET="192.168.2.0/24"            # Internal
Users/RoadWarriors subnet
APP_NET="192.168.3.0/24"        # Application/Database subnet
IDS_NET="192.168.255.0/24"              # IDS subnet

DMZ_SEG="172.16.1.0/27"               # DMZ segment
WEBC_SEG="172.16.1.32/27"       # Web Cluster segment
INTERM_SEG="172.16.1.240/29"          # Intermediary segment
PERIM_SEG="172.16.1.248/29"           # Perimeter segment

AS_NS="172.16.1.1"              # A_NS host - This is the
primary DNS Server
AC_NS="172.16.1.2"              # DJBDNS requires a different
IP address if you want to run a DNS server and a Resolver on
the same machine
```

```
BS_NS="172.16.1.3"              # B_NS host - This is the
secondary DNS server
BC_NS="172.16.1.4"              # DJBDNS requires a different
IP address if you want to run a DNS server and a Resolver on
the same machine

A_MX="172.16.1.5"               # A_MX host - This is the
primary Mail Server
B_MX="172.16.1.6"               # B_MX host - This is the
secondary Mail Server

WWW1="172.16.1.33"
WWW2="172.16.1.34"
WWW3="172.16.1.35"
WWW4="172.16.1.36"

NTP="192.168.1.1"                     # IP address of tick
(NTP server)
LOGHOST="192.168.1.3"                 # IP address of goiaba
(Loghost)
INT_NC="172.16.1.241"                 # IP address of graviola
(Internal DNS cache/resolver)
INT_MX="192.168.1.4"                  # IP address of int.mx
(Internal Mail server)
BACKUP="192.168.1.9"                  # IP address of cocada
(Backup server)
SNMP="192.168.1.10"             # IP address of tapioca
(SNMP/MRTG station)
PROXY="172.16.1.241"                  # IP address of doritos
(HTTP/S and FTP Squid proxy server)

APPS1="192.168.3.8"             # Application Server 1
APPS2="192.168.3.9"             # Application Server 2

# xml.rhn.redhat.com 66.187.232.101
RHN="66.187.232.101"
```

2.2.2 - Global configuration

In this section, it is defined the firewall policy and all chains needed. The default policy is deny everything, rules are constructed to allow in or out a specific traffic. Extra chains are created to facilitate the System Administrator's job. The INPUT chain is used to control traffic to the firewall, OUTPUT chain is used to control traffic from the firewall and FORWARD to control traffic traversing the firewall.

```
$IPTABLES -P INPUT DROP
$IPTABLES -P OUTPUT DROP
$IPTABLES -P FORWARD DROP

$IPTABLES -N LOCAL_INPUT
$IPTABLES -N LOCAL_OUTPUT
$IPTABLES -N LOCAL_FORWARD
```

```
$IPTABLES -N AUTO_INPUT
$IPTABLES -N AUTO_OUTPUT
$IPTABLES -N AUTO_FORWARD

$IPTABLES -N LOGDROP
$IPTABLES -N LOGREJECT
$IPTABLES -N LOGC

$IPTABLES -N FWD_DMZ
$IPTABLES -N FWD_WEBC
$IPTABLES -N FWD_INTERM
$IPTABLES -N FWD_PERIM

$IPTABLES -N ICMPC

# Chain INPUT
$IPTABLES -A INPUT -s 0/0 -d 0/0 -j LOCAL_INPUT
$IPTABLES -A INPUT -s 0/0 -d 0/0 -j AUTO_INPUT
$IPTABLES -A INPUT -s 0/0 -d 0/0 -j LOGDROP

# Chain OUTPUT
$IPTABLES -A OUTPUT -s 0/0 -d 0/0 -j LOCAL_OUTPUT
$IPTABLES -A OUTPUT -s 0/0 -d 0/0 -j AUTO_OUTPUT
$IPTABLES -A OUTPUT -s 0/0 -d 0/0 -j LOGDROP

# Chain FORWARD
$IPTABLES -A FORWARD -s 0/0 -d 0/0 -j LOCAL_FORWARD
$IPTABLES -A FORWARD -s 0/0 -d 0/0 -j AUTO_FORWARD
$IPTABLES -A FORWARD -s 0/0 -d 0/0 -j LOGDROP
```

2.2.3 - Rules Definition

Here is defined the rules used by all chains defined above.

```
# Chain LOCAL_INPUT - This chain is used to define rules for
services provided by the firewall itself.

# SSH - accepts SSH connections from Services-Management
segment
$IPTABLES -A LOCAL_INPUT -i $INTERM_IFACE -p tcp -m state --
state NEW -s $MNGT_NET --sport 1024:65535 -d $LARANJA_INTERM -
-dport 22 -j ACCEPT

# SNMP - Accepts SNMP polling from the SNMP management station
$IPTABLES -A LOCAL_INPUT -i $INTERM_IFACE -p udp -m state --
state NEW -s $SNMP --sport 1024:65535 -d $LARANJA_INTERM --
dport 161 -j ACCEPT

# BACKUP - Accepts connection from the Backup server
```

```
$IPTABLES -A LOCAL_INPUT -i $INTERM_IFACE -p tcp -m state --
state NEW -s $BACKUP --sport 1024:65535 -d $LARANJA_INTERM --
dport 8192:8193 -j ACCEPT

# Chain AUTO_INPUT - This chain is used to define default
INPUT rules.
$IPTABLES -A AUTO_INPUT -m state --state RELATED,ESTABLISHED -
j ACCEPT

# The loopback interface should run free and wild
$IPTABLES -A AUTO_INPUT -i $LOOPBACK_IFACE -s 127.0.0.0/8 to
127.0.0.0/8 -j ACCEPT

# Allow traceroute to the firewall.
$IPTABLES -A AUTO_INPUT -m state --state NEW -s 0/0 -d 0/0 --
dport 33434:33690 -j ACCEPT

# Allow some ICMP traffic to the firewall.
$IPTABLES -A AUTO_INPUT -p icmp -s 0/0 -d 0/0 -j ICMPC

# Chain LOCAL_OUTPUT - This chain is used to define rules for
services used by the firewall

# DNS - Allow the firewall to make queries on DNS
cache/resolver at DMZ
$IPTABLES -A LOCAL_OUTPUT -p udp -m state --state NEW -s
$LARANJA_DMZ -d $AC_NS --dport 53 -j ACCEPT
$IPTABLES -A LOCAL_OUTPUT -p udp -m state --state NEW -s
$LARANJA_DMZ -d $BC_NS --dport 53 -j ACCEPT

# SYSLOG - Allow the firewall to send its logs to LOGHOST
$IPTABLES -A LOCAL_OUTPUT -p udp -m state --state NEW -s
$LARANJA_INTERM --sport 1024:65535 -d $LOGHOST --dport 514 -j
ACCEPT

# NTP - Allow the firewall to synchronize its internal clock
with GIAC's NTP server
$IPTABLES -A LOCAL_OUTPUT -p udp -m state --state NEW -s
$LARANJA_INTERM --sport 123 -d $NTP --dport 123 -j ACCEPT

# BACKUP - Allow the Backup Agent to communicate with Backup
Server
$IPTABLES -A LOCAL_OUTPUT -p tcp -m state --state NEW -s
$LARANJA_INTERM --sport 1024:65535 -d $BACKUP --dport 6101 -j
ACCEPT

# RHN - Allow the firewall to get updates from Red Hat Network
$IPTABLES -A LOCAL_OUTPUT -p tcp -m state --state NEW -s
$LARANJA_INTERM --sport 1024:65535 -d $RHN --dport 443 -j
ACCEPT
```

```
# Chain AUTO_OUTPUT - This chain is used to define default
OUTPUT rules
$IPTABLES -A AUTO_OUTPUT -m state --state RELATED,ESTABLISHED
-j ACCEPT

# The loopback interface should run free and wild
$IPTABLES -A AUTO_OUTPUT -o $LOOPBACK_IFACE -s 127.0.0.0/8 to
127.0.0.0/8 -j ACCEPT

# Allow traceroute from the firewall
$IPTABLES -A AUTO_OUTPUT -m state --state NEW -s 0/0 -d 0/0 --
dport 33434:33690 -j ACCEPT

# Allow some ICMP traffic from the firewall
$IPTABLES -A AUTO_OUTPUT -p icmp -s 0/0 -d 0/0 -j ICMPC

# Chain LOCAL_FORWARD -
$IPTABLES -A LOCAL_FORWARD -s 0/0 -d 0/0 -j $FWD_DMZ
$IPTABLES -A LOCAL_FORWARD -s 0/0 -d 0/0 -j $FWD_WEBC
$IPTABLES -A LOCAL_FORWARD -s 0/0 -d 0/0 -j $FWD_INT
$IPTABLES -A LOCAL_FORWARD -s 0/0 -d 0/0 -j $FWD_PERIM

# Chain AUTO_FORWARD - This chain is used to define default
FORWARD rules
$IPTABLES -A AUTO_FORWARD -m state --state RELATED,ESTABLISHED
-j ACCEPT

# Allow traceroute traverse the firewall
$IPTABLES -A AUTO_FORWARD -m state --state NEW -s 0/0 -d 0/0 -
-dport 33434:33690 -j ACCEPT

# Allow some ICMP messages traverse the firewall
$IPTABLES -A AUTO_FORWARD -p icmp -s 0/0 -d 0/0 -j ICMPC

# Chain FWD_DMZ - All rules related to DMZ

# Outside GIAC -> DMZ

# DNS server - Allow packets to DNS server
$IPTABLES -A FWD_DMZ -i $PERIM_IFACE -o $DMZ_IFACE -m state --
state NEW -p udp -s 0/0 -d $AS_NS --dport 53 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $PERIM_IFACE -o $DMZ_IFACE -m state --
state NEW -p udp -s 0/0 -d $BS_NS --dport 53 -j ACCEPT

# SMTP server - Allow packets to SMTP server
$IPTABLES -A FWD_DMZ -i $PERIM_IFACE -o $DMZ_IFACE -m state --
state NEW -p tcp -s 0/0 -d $A_MX --dport 25 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $PERIM_IFACE -o $DMZ_IFACE -m state --
state NEW -p tcp -s 0/0 -d $B_MX --dport 25 -j ACCEPT

# Web Cluster -> DMZ
# DNS cache - Allow DNS queries from Web Cluster segment
$IPTABLES -A FWD_DMZ -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p udp -s $WWW1 -d $AC_NS --dport 53 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p udp -s $WWW1 -d $BC_NS --dport 53 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p udp -s $WWW2 -d $AC_NS --dport 53 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p udp -s $WWW2 -d $BC_NS --dport 53 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p udp -s $WWW3 -d $AC_NS --dport 53 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p udp -s $WWW3 -d $BC_NS --dport 53 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p udp -s $WWW4 -d $AC_NS --dport 53 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p udp -s $WWW4 -d $BC_NS --dport 53 -j ACCEPT

# SMTP - Allow SMTP traffic from Web Cluster segment to A_MX
and B_MX
$IPTABLES -A FWD_DMZ -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p tcp -s $WWW1 -d $A_MX --dport 25 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p tcp -s $WWW1 -d $B_MX --dport 25 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p tcp -s $WWW2 -d $A_MX --dport 25 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p tcp -s $WWW2 -d $B_MX --dport 25 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p tcp -s $WWW3 -d $A_MX --dport 25 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p tcp -s $WWW3 -d $B_MX --dport 25 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p tcp -s $WWW4 -d $A_MX --dport 25 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p tcp -s $WWW4 -d $B_MX --dport 25 -j ACCEPT

# DMZ -> Intermediary (and internal network)

# SMTP - Allow A_MX and B_MX send SMTP traffic to INT_MX
$IPTABLES -A FWD_DMZ -i $DMZ_IFACE -o $INTERM_IFACE -m state -
-state NEW -p tcp -s $A_MX -d $INT_MX --dport 25 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $DMZ_IFACE -o $INTERM_IFACE -m state -
-state NEW -p tcp -s $B_MX -d $INT_MX --dport 25 -j ACCEPT

# SYSLOG - Allow all machines on DMZ to send their logs to
LOGHOST
$IPTABLES -A FWD_DMZ -i $DMZ_IFACE -o $INTERM_IFACE -m state -
-state NEW -p udp -s $AS_NS --sport 1024:65535 -d $LOGHOST --
dport 514 -j ACCEPT
```

```
$IPTABLES -A FWD_DMZ -i $DMZ_IFACE -o $INTERM_IFACE -m state -
-state NEW -p udp -s $BS_NS --sport 1024:65535 -d $LOGHOST --
dport 514 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $DMZ_IFACE -o $INTERM_IFACE -m state -
-state NEW -p udp -s $A_MX --sport 1024:65535 -d $LOGHOST --
dport 514 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $DMZ_IFACE -o $INTERM_IFACE -m state -
-state NEW -p udp -s $B_MX --sport 1024:65535 -d $LOGHOST --
dport 514 -j ACCEPT

# NTP - Allow all machines on DMZ to synchronize their clocks
with the internal NTP server
$IPTABLES -A FWD_DMZ -i $DMZ_IFACE -o $INTERM_IFACE -m state -
-state NEW -p udp -s $AS_NS --sport 123 -d $NTP --dport 123 -j
ACCEPT
$IPTABLES -A FWD_DMZ -i $DMZ_IFACE -o $INTERM_IFACE -m state -
-state NEW -p udp -s $BS_NS --sport 123 -d $NTP --dport 123 -j
ACCEPT
$IPTABLES -A FWD_DMZ -i $DMZ_IFACE -o $INTERM_IFACE -m state -
-state NEW -p udp -s $A_MX --sport 123 -d $NTP --dport 123 -j
ACCEPT
$IPTABLES -A FWD_DMZ -i $DMZ_IFACE -o $INTERM_IFACE -m state -
-state NEW -p udp -s $B_MX --sport 123 -d $NTP --dport 123 -j
ACCEPT

# BACKUP - Allow all machines on DMZ to communicate with the
Backup serve
$IPTABLES -A FWD_DMZ -i $DMZ_IFACE -o $INTERM_IFACE -m state -
-state NEW -p tcp -s $AS_NS --sport 1024:65535 -d $BACKUP --
dport 6101 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $DMZ_IFACE -o $INTERM_IFACE -m state -
-state NEW -p tcp -s $BS_NS --sport 1024:65535 -d $BACKUP --
dport 6101 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $DMZ_IFACE -o $INTERM_IFACE -m state -
-state NEW -p tcp -s $A_MX --sport 1024:65535 -d $BACKUP --
dport 6101 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $DMZ_IFACE -o $INTERM_IFACE -m state -
-state NEW -p tcp -s $B_MX --sport 1024:65535 -d $BACKUP --
dport 6101 -j ACCEPT

# Intermediary (and internal network) -> DMZ

# SSH - accepts SSH connections from Services-Management
segment
$IPTABLES -A FWD_DMZ -i $INTERM_IFACE -p tcp -m state --state
NEW -s $MNGT_NET --sport 1024:65535 -d $A_MX --dport 22 -j
ACCEPT
$IPTABLES -A FWD_DMZ -i $INTERM_IFACE -p tcp -m state --state
NEW -s $MNGT_NET --sport 1024:65535 -d $B_MX --dport 22 -j
ACCEPT
```

```
$IPTABLES -A FWD_DMZ -i $INTERM_IFACE -p tcp -m state --state
NEW -s $MNGT_NET --sport 1024:65535 -d $A_NS --dport 22 -j
ACCEPT
$IPTABLES -A FWD_DMZ -i $INTERM_IFACE -p tcp -m state --state
NEW -s $MNGT_NET --sport 1024:65535 -d $B_NS --dport 22 -j
ACCEPT

#SMTP - Allow SMTP traffic from INT_MX to A_MX and B_MX
$IPTABLES -A FWD_DMZ -i $INTERM_IFACE -o $DMZ_IFACE -m state -
-state NEW -p tcp -s $INT_MX -d $A_MX --dport 25 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $INTERM_IFACE -o $DMZ_IFACE -m state -
-state NEW -p tcp -s $INT_MX -d $B_MX --dport 25 -j ACCEPT

# SNMP - Allow SNMP polling from tapipoca to all machines on
DMZ
$IPTABLES -A FWD_DMZ -i $INTERM_IFACE -o $DMZ_IFACE -m state -
-state NEW -p udp -s $SNMP --sport 1024:65535 -d $AS_NS --
dport 161 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $INTERM_IFACE -o $DMZ_IFACE -m state -
-state NEW -p udp -s $SNMP --sport 1024:65535 -d $BS_NS --
dport 161 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $INTERM_IFACE -o $DMZ_IFACE -m state -
-state NEW -p udp -s $SNMP --sport 1024:65535 -d $A_MX --dport
161 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $INTERM_IFACE -o $DMZ_IFACE -m state -
-state NEW -p udp -s $SNMP --sport 1024:65535 -d $B_MX --dport
161 -j ACCEPT

# BACKUP - Allow Backup server to communicate with all
machines on DMZ
$IPTABLES -A FWD_DMZ -i $INTERM_IFACE -o $DMZ_IFACE -m state -
-state NEW -p tcp -s $BACKUP --sport 1024:65535 -d $AS_NS --
dport 8192:8193 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $INTERM_IFACE -o $DMZ_IFACE -m state -
-state NEW -p tcp -s $BACKUP --sport 1024:65535 -d $BS_NS --
dport 8192:8193 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $INTERM_IFACE -o $DMZ_IFACE -m state -
-state NEW -p tcp -s $BACKUP --sport 1024:65535 -d $A_MX --
dport 8192:8193 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $INTERM_IFACE -o $DMZ_IFACE -m state -
-state NEW -p tcp -s $BACKUP --sport 1024:65535 -d $B_MX --
dport 8192:8193 -j ACCEPT

# DMZ -> outside GIAC

# DNS cache/resolver - Allow the DNS resolver to ask other DNS
servers
$IPTABLES -A FWD_DMZ -i $DMZ_IFACE -o $PERIM_IFACE -m state --
state NEW -p udp -s $AC_NS -d 0/0 --dport 53 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $DMZ_IFACE -o $PERIM_IFACE -m state --
state NEW -p udp -s $BC_NS -d 0/0 --dport 53 -j ACCEPT
```

```
# SMTP - Allow A_MX and B_X to send email to other mail
servers
$IPTABLES -A FWD_DMZ -i $DMZ_IFACE -o $PERIM_IFACE -m state --
state NEW -p tcp -s $A_MX -d 0/0 --dport 25 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $DMZ_IFACE -o $PERIM_IFACE -m state --
state NEW -p tcp -s $B_MX -d 0/0 --dport 25 -j ACCEPT

# RedHat Network - Allow all machines on DMZ to get updates
from Red Hat Network
$IPTABLES -A FWD_DMZ -i $DMZ_IFACE -o $PERIM_IFACE -m state --
state NEW -p tcp -s $AS_NS -d $RHN --dport 443 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $DMZ_IFACE -o $PERIM_IFACE -m state --
state NEW -p tcp -s $BS_NS -d $RHN --dport 443 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $DMZ_IFACE -o $PERIM_IFACE -m state --
state NEW -p tcp -s $A_MX -d $RHN --dport 443 -j ACCEPT
$IPTABLES -A FWD_DMZ -i $DMZ_IFACE -o $PERIM_IFACE -m state --
state NEW -p tcp -s $B_MX -d $RHN --dport 443 -j ACCEPT

# Chain FWD_WEBC - All rules related with Web Cluster segment
are defined here

# outside GIAC -> WEBC

# HTTP - Allow customers to access the main website using HTTP
$IPTABLES -A FWD_WEBC -i $PERIM_IFACE -o $WEBC_IFACE -m state
--state NEW -p tcp -s 0/0 -d $WWW1 --dport 80 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $PERIM_IFACE -o $WEBC_IFACE -m state
--state NEW -p tcp -s 0/0 -d $WWW2 --dport 80 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $PERIM_IFACE -o $WEBC_IFACE -m state
--state NEW -p tcp -s 0/0 -d $WWW3 --dport 80 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $PERIM_IFACE -o $WEBC_IFACE -m state
--state NEW -p tcp -s 0/0 -d $WWW4 --dport 80 -j ACCEPT

# HTTPS - Allow customers to access the main website using
HTTPS
$IPTABLES -A FWD_WEBC -i $PERIM_IFACE -o $WEBC_IFACE -m state
--state NEW -p tcp -s 0/0 -d $WWW1 --dport 443 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $PERIM_IFACE -o $WEBC_IFACE -m state
--state NEW -p tcp -s 0/0 -d $WWW2 --dport 443 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $PERIM_IFACE -o $WEBC_IFACE -m state
--state NEW -p tcp -s 0/0 -d $WWW3 --dport 443 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $PERIM_IFACE -o $WEBC_IFACE -m state
--state NEW -p tcp -s 0/0 -d $WWW4 --dport 443 -j ACCEPT

# Intermediary (and internal network) -> Web Cluster

# SSH - accepts SSH connections from Services-Management
segment
```

```
$IPTABLES -A FWD_WEBC -i $INTERM_IFACE -p tcp -m state --state
NEW -s $MNGT_NET --sport 1024:65535 -d $WWW1 --dport 22 -j
ACCEPT
$IPTABLES -A FWD_WEBC -i $INTERM_IFACE -p tcp -m state --state
NEW -s $MNGT_NET --sport 1024:65535 -d $WWW2 --dport 22 -j
ACCEPT
$IPTABLES -A FWD_WEBC -i $INTERM_IFACE -p tcp -m state --state
NEW -s $MNGT_NET --sport 1024:65535 -d $WWW3 --dport 22 -j
ACCEPT
$IPTABLES -A FWD_WEBC -i $INTERM_IFACE -p tcp -m state --state
NEW -s $MNGT_NET --sport 1024:65535 -d $WWW4 --dport 22 -j
ACCEPT

# HTTP - Allow internal users to access main the website using
HTTP
$IPTABLES -A FWD_WEBC -i $INTERM_IFACE -o $WEBC_IFACE -m state
--state NEW -p tcp -s $USERS_NET --sport 1024:65535 -d $WWW1 -
-dport 80 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $INTERM_IFACE -o $WEBC_IFACE -m state
--state NEW -p tcp -s $USERS_NET --sport 1024:65535 -d $WWW2 -
-dport 80 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $INTERM_IFACE -o $WEBC_IFACE -m state
--state NEW -p tcp -s $USERS_NET --sport 1024:65535 -d $WWW3 -
-dport 80 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $INTERM_IFACE -o $WEBC_IFACE -m state
--state NEW -p tcp -s $USERS_NET --sport 1024:65535 -d $WWW4 -
-dport 80 -j ACCEPT

# HTTPS - Allow internal users to access the main website
using HTTPS
$IPTABLES -A FWD_WEBC -i $INTERM_IFACE -o $WEBC_IFACE -m state
--state NEW -p tcp -s $USERS_NET --sport 1024:65535 -d $WWW1 -
-dport 443 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $INTERM_IFACE -o $WEBC_IFACE -m state
--state NEW -p tcp -s $USERS_NET --sport 1024:65535 -d $WWW2 -
-dport 443 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $INTERM_IFACE -o $WEBC_IFACE -m state
--state NEW -p tcp -s $USERS_NET --sport 1024:65535 -d $WWW3 -
-dport 443 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $INTERM_IFACE -o $WEBC_IFACE -m state
--state NEW -p tcp -s $USERS_NET --sport 1024:65535 -d $WWW4 -
-dport 443 -j ACCEPT

# BACKUP - Allow the backup server to communicate with all
machines on Web Cluster
$IPTABLES -A FWD_WEBC -i $INTERM_IFACE -o $WEBC_IFACE -m state
--state NEW -p tcp -s $BACKUP --sport 1024:65535 -d $WWW1 --
dport 8192:8193 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $INTERM_IFACE -o $WEBC_IFACE -m state
--state NEW -p tcp -s $BACKUP --sport 1024:65535 -d $WWW2 --
dport 8192:8193 -j ACCEPT
```

```
$IPTABLES -A FWD_WEBC -i $INTERM_IFACE -o $WEBC_IFACE -m state
--state NEW -p tcp -s $BACKUP --sport 1024:65535 -d $WWW3 --
dport 8192:8193 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $INTERM_IFACE -o $WEBC_IFACE -m state
--state NEW -p tcp -s $BACKUP --sport 1024:65535 -d $WWW4 --
dport 8192:8193 -j ACCEPT

# SNMP - Allow SNMP polling from tapipoca to all machines on
Web Cluster
$IPTABLES -A FWD_WEBC -i $INTERM_IFACE -o $WEBC_IFACE -m state
--state NEW -p udp -s $SNMP --sport 1024:65535 -d $WWW1 --
dport 161 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $INTERM_IFACE -o $WEBC_IFACE -m state
--state NEW -p udp -s $SNMP --sport 1024:65535 -d $WWW2 --
dport 161 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $INTERM_IFACE -o $WEBC_IFACE -m state
--state NEW -p udp -s $SNMP --sport 1024:65535 -d $WWW3 --
dport 161 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $INTERM_IFACE -o $WEBC_IFACE -m state
--state NEW -p udp -s $SNMP --sport 1024:65535 -d $WWW4 --
dport 161 -j ACCEPT

# Web Cluster -> Intermediary (and internal network)

# SYSLOG - Allow all machines on Web Cluster to send their
logs to LOGHOST
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $INTERM_IFACE -m state
--state NEW -p udp -s $WWW1 --sport 1024:65535 -d $LOGHOST --
dport 514 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $INTERM_IFACE -m state
--state NEW -p udp -s $WWW2 --sport 1024:65535 -d $LOGHOST --
dport 514 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $INTERM_IFACE -m state
--state NEW -p udp -s $WWW3 --sport 1024:65535 -d $LOGHOST --
dport 514 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $INTERM_IFACE -m state
--state NEW -p udp -s $WWW4 --sport 1024:65535 -d $LOGHOST --
dport 514 -j ACCEPT

# NTP - Allow all machines on Web Cluster to synchronize their
clocks with the NTP server
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $INTERM_IFACE -m state
--state NEW -p udp -s $WWW1 --sport 123 -d $NTP --dport 123 -j
ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $INTERM_IFACE -m state
--state NEW -p udp -s $WWW2 --sport 123 -d $NTP --dport 123 -j
ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $INTERM_IFACE -m state
--state NEW -p udp -s $WWW3 --sport 123 -d $NTP --dport 123 -j
ACCEPT
```

```
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $INTERM_IFACE -m state
--state NEW -p udp -s $WWW4 --sport 123 -d $NTP --dport 123 -j
ACCEPT

# BACKUP - Allow all machines on Web Cluster to communicate
with the Backup server
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $INTERM_IFACE -m state
--state NEW -p tcp -s $WWW1 --sport 1024:65535 -d $BACKUP --
dport 6101 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $INTERM_IFACE -m state
--state NEW -p tcp -s $WWW2 --sport 1024:65535 -d $BACKUP --
dport 6101 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $INTERM_IFACE -m state
--state NEW -p tcp -s $WWW3 --sport 1024:65535 -d $BACKUP --
dport 6101 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $INTERM_IFACE -m state
--state NEW -p tcp -s $WWW4 --sport 1024:65535 -d $BACKUP --
dport 6101 -j ACCEPT

# APP servers - Allow all machines on Web Cluster to
communicate with the Application Servers
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $INTERM_IFACE -m state
--state NEW -p tcp -s $WWW1 --sport 1024:65535 -d $APPS1 --
dport 8443 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $INTERM_IFACE -m state
--state NEW -p tcp -s $WWW2 --sport 1024:65535 -d $APPS1 --
dport 8443 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $INTERM_IFACE -m state
--state NEW -p tcp -s $WWW3 --sport 1024:65535 -d $APPS1 --
dport 8443 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $INTERM_IFACE -m state
--state NEW -p tcp -s $WWW4 --sport 1024:65535 -d $APPS1 --
dport 8443 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $INTERM_IFACE -m state
--state NEW -p tcp -s $WWW1 --sport 1024:65535 -d $APPS2 --
dport 8443 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $INTERM_IFACE -m state
--state NEW -p tcp -s $WWW2 --sport 1024:65535 -d $APPS2 --
dport 8443 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $INTERM_IFACE -m state
--state NEW -p tcp -s $WWW3 --sport 1024:65535 -d $APPS2 --
dport 8443 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $INTERM_IFACE -m state
--state NEW -p tcp -s $WWW4 --sport 1024:65535 -d $APPS2 --
dport 8443 -j ACCEPT

# Web Cluster -> DMZ

# DNS - Allow all machines on Web Cluster to send DNS queries
to DNS servers at DMZ
```

```
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p udp -s $WWW1 -d $AC_NS --dport 53 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p udp -s $WWW1 -d $BC_NS --dport 53 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p udp -s $WWW2 -d $AC_NS --dport 53 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p udp -s $WWW2 -d $BC_NS --dport 53 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p udp -s $WWW3 -d $AC_NS --dport 53 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p udp -s $WWW3 -d $BC_NS --dport 53 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p udp -s $WWW4 -d $AC_NS --dport 53 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p udp -s $WWW4 -d $BC_NS --dport 53 -j ACCEPT

# SMTP - Allow all machines on Web Cluster to send email to
SMTP servers at DMZ (A_MX and B_MX)
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p tcp -s $WWW1 -d $A_MX --dport 25 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p tcp -s $WWW1 -d $B_MX --dport 25 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p tcp -s $WWW2 -d $A_MX --dport 25 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p tcp -s $WWW2 -d $B_MX --dport 25 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p tcp -s $WWW3 -d $A_MX --dport 25 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p tcp -s $WWW3 -d $B_MX --dport 25 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p tcp -s $WWW4 -d $A_MX --dport 25 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $DMZ_IFACE -m state --
state NEW -p tcp -s $WWW4 -d $B_MX --dport 25 -j ACCEPT

# Web Cluster -> outside GIAC

# RHN - Allow all machines on Web Cluster to get updates from
Red Hat Network
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $PERIM_IFACE -m state
--state NEW -p tcp -s $WWW1 -d $RHN --dport 443 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $PERIM_IFACE -m state
--state NEW -p tcp -s $WWW2 -d $RHN --dport 443 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $PERIM_IFACE -m state
--state NEW -p tcp -s $WWW3 -d $RHN --dport 443 -j ACCEPT
$IPTABLES -A FWD_WEBC -i $WEBC_IFACE -o $PERIM_IFACE -m state
--state NEW -p tcp -s $WWW4 -d $RHN --dport 443 -j ACCEPT

# Chain FWD_INTERM - This chain is used to define rules for
the interface facing the intermediary (and internal) network.
```

```
# Intermediary (and internal network) -> OUTSIDE GIAC

# DNS - Allow the DNS cache/resolver communicate with DNS
servers outside GIAC
$IPTABLES -A FWD_INTERM -i $INTERM_IFACE -o $PERIM_IFACE -p
udp -m state --state NEW -s $INT_NC --sport 1024:65535 -d 0/0
--dport 53 -j ACCEPT

# HTTP/S PROXY - Allow the HTTP/S and FTP proxy coomunicate
with HTTP/S servers over the web
$IPTABLES -A FWD_INTERM -i $INTERM_IFACE -o $PERIM_IFACE -p
tcp -m state --state NEW -s $PROXY --sport 1024:65535 -d 0/0 -
-dport 80 -j ACCEPT
$IPTABLES -A FWD_INTERM -i $INTERM_IFACE -o $PERIM_IFACE -p
tcp -m state --state NEW -s $PROXY --sport 1024:65535 -d 0/0 -
-dport 443 -j ACCEPT

# ACTIVE/PASSIVE FTP client PROXY

$IPTABLES -A FWD_INTERM -p tcp -m state --state
ESTABLISHED,RELATED -s 0/0 --sport 20 -d $PROXY --dport
1024:65535 -j ACCEPT
$IPTABLES -A FWD_INTERM -p tcp -m state --state
ESTABLISHED,RELATED -s 0/0 --sport 1024:65535 -d $PROXY --
dport 1024:65535 -j ACCEPT
$IPTABLES -A FWD_INTERM -p tcp -m state --state
ESTABLISHED,RELATED -s 0/0 --sport 21 -d $PROXY --dport
1024:65535 -j ACCEPT
$IPTABLES -A FWD_INTERM -p tcp -m state --state NEW -s $PROXY
--sport 1024:65535 -d 0/0 --dport 21 -j ACCEPT
$IPTABLES -A FWD_INTERM -p tcp -m state --state
ESTABLISHED,RELATED -s $PROXY --sport 1024:65535 -d 0/0 --
dport 1024:65535

# Chain FWD_PERIM - This chain is used to define rules for the
interface facing the perimeter network.

# Block all telnet attempts
$IPTABLES -A FWD_PERIM -o $PERIM_IFACE -p tcp -s 0/0 -d 0/0 --
dport 23 -j LOGDROP

# With too vulnerabilities and worms related with Microsoft
software, GIAC Enterprises decided to block some Microsoft
Services
$IPTABLES -A FWD_PERIM -o $PERIM_IFACE -p tcp -s 0/0 -d 0/0 --
dport 135:139 -j LOGDROP
$IPTABLES -A FWD_PERIM -o $PERIM_IFACE -p udp -s 0/0 -d 0/0 --
dport 135:139 -j LOGDROP
$IPTABLES -A FWD_PERIM -o $PERIM_IFACE -p tcp -s 0/0 -d 0/0 --
dport 445 -j LOGDROP
```

```
# Chain ICMPC - This chain is used to ICMP types allowed
IN/OUT GIAC Enterprises

# Allow ICMP ECHO
$IPTABLES -A ICMPC -p icmp --icmp-type 8/0 -s 0/0 -d 0/0 -m
limit --limit 1/s -j ACCEPT
# Allow ICMP ECHO REPLY
$IPTABLES -A ICMPC -p icmp --icmp-type 0/0 -s 0/0 -d 0/0 -m
limit --limit 1/s -j ACCEPT
# Allow ICMP UNREACH_NEEDFRAG - used by PATH MTU
$IPTABLES -A ICMPC -p icmp --icmp-type 3/4 -s 0/0 -d 0/0 -j
ACCEPT
# Allow Source Quench
$IPTABLES -A ICMPC -p icmp --icmp-type 4/0 -s 0/0 -d 0/0 -j
ACCEPT
# Allow Time Exceeded (TTL expired in transit)
$IPTABLES -A ICMPC -p icmp --icmp-type 11/0 -s 0/0 -d 0/0 -j
ACCEPT
# LOG and DROP other ICMP messages
$IPTABLES -A ICMPC -p icmp -s 0/0 -d 0/0 -j LOGDROP

# Chain LOGDROP - Jump to LOGC and then drop the packet
$IPTABLES -A LOGDROP -s 0/0 -d 0/0 -j LOGC
$IPTABLES -A LOGDROP -s 0/0 -d 0/0 -j DROP

# LOGREJECT - Jump to LOGC and reject the packet sending back
a ICMP message
$IPTABLES -A LOGREJECT -s 0/0 -d 0/0 -j LOGC
$IPTABLES -A LOGREJECT -s 0/0 -d 0/0 -j REJECT

#Chain LOGC (Log Chain) - Just log tcp, udp, icmp, ESP, HA and
fragmented packets
$IPTABLES -A LOGC -p tcp -s 0/0 -d 0/0 -j LOG
$IPTABLES -A LOGC -p udp -s 0/0 -d 0/0 -j LOG
$IPTABLES -A LOGC -p icmp -s 0/0 -d 0/0 -j LOG
$IPTABLES -A LOGC -p 50 -s 0/0 -d 0/0 -j LOG
$IPTABLES -A LOGC -p 51 -s 0/0 -d 0/0 -j LOG
$IPTABLES -A LOGC -f -s 0/0 -d 0/0 -j LOG
```

2.3 - Internal Firewall

The internal firewall is the third and last layer of defense of GIAC Enterprises. limao is a FreeBSD-4.8STABLE running IPFILTER as a packet filter. Its main function is to protect itself and all internal networks/segments (Management/Services 192.168.1.0/24, Internal Users/RoadWarriors 192.168.2.0/24, Application/Database 192.168.3.0/24 and IDS 192.168.255.0/24). The communications of every component on internal network is controlled by this firewall.

2.3.1 - IPFILTER firewall rules /etc/ipf.rules

```
# $Id: /etc/ipf.rules, v 0.04 2003.09.25 51:23 alexcm Exp $
# Interface definition
#
# fxp0 - external
# fxp1 - services/management
# fxp2 - internal/VPN/Wi-Fi users
# fxp3 - database/web applications
# fxp4 - IDS

# the loopback interface should run free and wild
pass in on lo0 all
pass out on lo0 all

# Block MARTIANS packets (fragmented and with IP options)
block in log quick from any to any with ipopts
block in log quick from any to any with short
block out log quick from any to any with ipopts
block out log quick from any to any with short

# ICMP
# Allow IN echo REPLY
pass in quick proto icmp any to any icmp-type 0
# Allow IN Network Unreachable
pass in quick proto icmp any to any icmp-type 3
# Allow IN Source Quench
pass in quick proto icmp any to any icmp-type 4
# Allow IN echo REQUEST
pass in quick proto icmp any to any icmp-type 8
# Allow IN TTL expired in transit
pass in quick proto icmp any to any icmp-type 11
# Block IN all ICMP messages
block in log quick proto icmp from any to any
# Allow OUT echo REPLY
pass out quick proto icmp any to any icmp-type 0
# Allow OUT Network Unreachable
pass out quick proto icmp any to any icmp-type 3
# Allow OUT Source Quench
pass out quick proto icmp any to any icmp-type 4
# Allow OUT echo REQUEST
pass out quick proto icmp any to any icmp-type 8
# Allow OUT TTL expired in transit
pass out quick proto icmp any to any icmp-type 11
block out log quick proto icmp from any to any

# Traceroute (UNIX) - Allow IN and OUT UNIX traceroute
pass in quick proto udp from any to any port 33434 >< 33690
pass out quick proto udp from any to any port 33434 >< 33690

# fxp0 IN
block in quick on fxp0 all head 10
```

```
# Some of Microsoft Windows protocols aren't allowed here
(192.168.0.0/16).
block in log quick on fxp0 proto tcp from any to
192.168.0.0/16 port 134 >< 140 group 10
block in log quick on fxp0 proto udp from any to
192.168.0.0/16 port 134 >< 140 group 10
block in log quick on fxp0 proto tcp from any to
192.168.0.0/16 port = 445 group 10

# this firewall does not offer any services to DMZ and outside
GIAC
block in log quick on fxp0 from any to 172.16.1.241/32 group
10

#
pass in on fxp0 all group 10

# fxp0 OUT
block out quick on fxp0 all head 20

# Some of Microsoft Windows protocols aren't allowed to leave
192.168.0.0/16
block out log quick on fxp0 proto tcp from 192.168.0.0/16 to
any port 134 >< 140 group 20
block out log quick on fxp0 proto udp from 192.168.0.0/16 to
any port 134 >< 140 group 20
block out log quick on fxp0 proto tcp from 192.168.0.0/16 to
any port = 445 group 20

#
pass out on fxp0 all group 20

# fxp1 IN (Services/Management -> somewhere)
block in quick on fxp1 all head 30

# tapioca SNMP management station - Allow SNMP polling on DMZ,
Web Cluster and Application/Database segments
pass in quick on fxp1 proto udp from 192.168.1.10/32 port >
1023 to 172.16.1.0/24 port = 161 keep state group 30
pass in quick on fxp1 proto udp from 192.168.1.10/32 port >
1023 to 192.168.3.0/24 port = 161 keep state group 30

# cocada Backup server - Allow Backup server communicate with
DMZ, Web Cluster and Application/Database segments
pass in quick on fxp1 proto tcp from 192.168.1.9/32 port >
1023 to 172.16.1.0/24 port 8191 >< 8194 flags S/SA keep state
group 30
pass in quick on fxp1 proto tcp from 192.168.1.9/32 port >
1023 to 192.168.3.0/24 port 8191 >< 8194 flags S/SA keep state
group 30
```

```
# graviola DNS cache/resolver for internal network - Allow the
cache/resolver to communicate with DNS servers outside GIAC
pass in quick on fxp1 proto udp from 192.168.1.8/32 port >
1023 to any port = 53 keep state group 30

# doritos HTTP/S and FTP proxy - Allow HTTP/S and FTP proxy
communicate with servers outside GIAC
pass in quick on fxp1 proto tcp from 192.168.1.6/32 port >
1023 to any port = 80 flags S/SA keep state group 30
pass in quick on fxp1 proto tcp from 192.168.1.6/32 port >
1023 to any port = 443 flags S/SA keep state group 30
pass in quick on fxp1 proto tcp from 192.168.1.6/32 port >
1023 to any port = 21 flags S/SA keep state group 30

# int.mx - Allow int.mx send deliver e-mail to a.mx and b.mx
(SMTP servers at DMZ)
pass in quick on fxp1 proto tcp from 192.168.1.4/32 port >
1023 to 172.16.1.5 port = 25 flags S/SA keep state group 30
pass in quick on fxp1 proto tcp from 192.168.1.4/32 port >
1023 to 172.16.1.6 port = 25 flags S/SA keep state group 30

# tick NTP server - Allow NTP server synchronize with other
NTP servers
pass in quick on fxp1 proto udp from 192.168.1.1/32 port = 123
to any port = 123 keep state group 30

# RedHat Network (xmlrpc.rhn.redhat.com) - Allow all machines
on Services/Management segment to get updates from Red Hat
Network
pass in quick on fxp1 proto tcp from 192.168.1.0/24 port >
1023 to 66.187.232.101/32 port = 443 flags S/SA keep state
group 30

# block everything else
block in log quick on fxp1 all group 30

# fxp1 OUT (somewhere -> Services/Management segment)
block out quick on fxp1 all head 40

# cocada Backup server - Allow Servers at DMZ, Web Cluster and
Database/Applications segments to communicate with Backup
server
pass out quick on fxp1 proto tcp from 172.16.1.0/24 port >
1023 to 192.168.1.9/32 port = 6101 flags S/SA keep state group
40
pass out quick on fxp1 proto tcp from 192.168.3.0/24 port >
1023 to 192.168.1.9/32 port = 6101 flags S/SA keep state group
40

# graviola DNS cache/resolver for internal network - Accept
DNS queries from Internal Users/RoadWarriors segment
```

```
pass out quick on fxp1 proto udp from 192.168.2.0/24 port >
1023 to 192.168.1.7/32 port = 53 keep state group 40
pass out quick on fxp1 proto udp from 192.168.3.0/24 port >
1023 to 192.168.1.7/32 port = 53 keep state group 40
pass out quick on fxp1 proto udp from 192.168.255.0/24 port >
1023 to 192.168.1.7/32 port = 53 keep state group 40

# doritos HTTP/S and FTP proxy - Allow Internal
Users/RoadWarriors to access the HTTP/S and FTP PROXY/CACHE
pass out quick on fxp1 proto tcp from 192.168.2.0/24 port >
1023 to 192.168.1.6/32 port = 3128 flags S/SA keep state group
40

# intranet (intranet server) - Allow Internal
Users/RoadWarriors to access GIAC's Intranet
pass out quick on fxp1 proto tcp from 192.168.2.0/24 port >
1023 to 192.168.1.5/32 port = 80 flags S/SA keep state group
40
pass out quick on fxp1 proto tcp from 192.168.2.0/24 port >
1023 to 192.168.1.5/32 port = 443 flags S/SA keep state group
40

# int.mx - Allow Internal Users to send e-mail using SMTP and
access their mail box using IMAP2, IMAPS and POP3S
pass out quick on fxp1 proto tcp from 192.168.2.0/24 port >
1023 to 192.168.1.4/32 port = 25 flags S/SA keep state group
40
pass out quick on fxp1 proto tcp from 192.168.2.0/24 port >
1023 to 192.168.1.4/32 port = 143 flags S/SA keep state group
40
pass out quick on fxp1 proto tcp from 192.168.2.0/24 port >
1023 to 192.168.1.4/32 port = 993 flags S/SA keep state group
40
pass out quick on fxp1 proto tcp from 192.168.2.0/24 port >
1023 to 192.168.1.4/32 port = 995 flags S/SA keep state group
40
# goiaba Syslog server - Allow all machines at DMZ to send
their logs to Loghost
pass out quick on fxp1 proto udp from 172.16.1.0/24 port >
1023 to 192.168.1.3/32 port = 514 keep state group 40

# quindim domain/wins/file/print/LDAP server - Allow Internal
Users/RoadWarriors to access Microsoft, LDAP and Printing
Services
pass out quick on fxp1 proto tcp from 192.168.3.0/24 port >
1023 to 192.168.1.2/32 port 134 >< 140 flags S/SA keep state
group 40
pass out quick on fxp1 proto udp from 192.168.3.0/24 port >
1023 to 192.168.1.2/32 port 134 >< 140 keep state group 40
```

```
pass out quick on fxp1 proto tcp from 192.168.3.0/24 port >
1023 to 192.168.1.2/32 port = 389 flags S/SA keep state group
40
pass out quick on fxp1 proto tcp from 192.168.3.0/24 port >
1023 to 192.168.1.2/32 port = 445 flags S/SA keep state group
40
pass out quick on fxp1 proto tcp from 192.168.3.0/24 port >
1023 to 192.168.1.2/32 port = 515 flags S/SA keep state group
40

# tick NTP server - Allow Internal Users/RoadWarriors, all
machines at DMZ, Web Cluster, IDS and Database/Applications
segment to synchronize their clocks with the internal NTP
server
pass out quick on fxp1 proto udp from 172.16.1.0/24 port = 123
to 192.168.1.1/32 port = 123 keep state group 40
pass out quick on fxp1 proto udp from 192.168.2.0/24 port =
123 to 192.168.1.1/32 port = 123 keep state group 40
pass out quick on fxp1 proto udp from 192.168.3.0/24 port =
123 to 192.168.1.1/32 port = 123 keep state group 40
pass out quick on fxp1 proto udp from 192.168.255.0/24 port =
123 to 192.168.1.1/32 port = 123 keep state group 40

# block everything else
block out log quick on fxp1 all group 40

# fxp2 IN (internal users -> somewhere)
block in quick on fxp2 all head 50

# DNS queries to graviola - Allow Internal Users/RoadWarriors
to make DNS queries to internal DNS cache/resolver
pass in quick on fxp2 proto udp from 192.168.2.0/24 port >
1023 to 192.168.1.8/32 port = 53 keep state group 50

# HTTP/S and FTP to doritos (proxy/cache) - Allow Internal
Users/RoadWarriors segment to access the HTTP/S and FTP proxy
pass in quick on fxp2 proto tcp from 192.168.2.0/24 port >
1023 to 192.168.1.6/32 port = 3128 flags S/SA keep state group
50

# Intranet access - Allow the Internal Users/RoadWarriors
segment to access the Intranet
pass in quick on fxp2 proto tcp from 192.168.2.0/24 port >
1023 to 192.168.1.5/32 port = 80 flags S/SA keep state group
50
pass in quick on fxp2 proto tcp from 192.168.2.0/24 port >
1023 to 192.168.1.5/32 port = 443 flags S/SA keep state group
50
```

```
# SMTP, IMAP2, IMAPS and POP3S access (int.mx) - Allow
Internal Users/RoadWarriors to send e-mail using SMTP and
accessing their mail box using IMAP2, IMAPS and POP3S
pass in quick on fxp2 proto tcp from 192.168.2.0/24 port >
1023 to 192.168.1.4/32 port = 25 flags S/SA keep state group
50
pass in quick on fxp2 proto tcp from 192.168.2.0/24 port >
1023 to 192.168.1.4/32 port = 143 flags S/SA keep state group
50
pass in quick on fxp2 proto tcp from 192.168.2.0/24 port >
1023 to 192.168.1.4/32 port = 993 flags S/SA keep state group
50
pass in quick on fxp2 proto tcp from 192.168.2.0/24 port >
1023 to 192.168.1.4/32 port = 995 flags S/SA keep state group
50

# domain/wins/file/print/LDAP access - Allow Internal
Users/RoadWarriors to access Microsoft, LDAP and Printing
services
pass in quick on fxp2 proto tcp from 192.168.2.0/24 port >
1023 to 192.168.1.2/32 port 134 >< 140 flags S/SA keep state
group 50
pass in quick on fxp2 proto udp from 192.168.2.0/24 port >
1023 to 192.168.1.2/32 port 134 >< 140 keep state group 50
pass in quick on fxp2 proto tcp from 192.168.2.0/24 port >
1023 to 192.168.1.2/32 port = 389 flags S/SA keep state group
50
pass in quick on fxp2 proto tcp from 192.168.2.0/24 port >
1023 to 192.168.1.2/32 port = 445 flags S/SA keep state group
50
pass in quick on fxp2 proto tcp from 192.168.2.0/24 port >
1023 to 192.168.1.2/32 port = 515 flags S/SA keep state group
50

# NTP access - Allow Internal Users/RoadWarriors to access the
internal NTP server
pass in quick on fxp2 proto udp from 192.168.2.0/24 port = 123
to 192.168.1.1/32 port = 123 keep state group 50

# RedHat Network (xmlrpc.rhn.redhat.com) - Allow Internal
Users/RoadWarriors to get updates from Red Hat Network
pass in quick on fxp2 proto tcp from 192.168.2.0/24 port >
1023 to 66.187.232.101/32 port = 443 flags S/SA keep state
group 50

# block everything else
block in log quick on fxp2 all group 50

# fxp2 OUT (somewhere -> internal users)
block out quick on fxp2 all head 60
```

```
# This segment does not offer services
block out log quick on fxp2 all group 60

# fxp3 IN (database/applications -> somewhere)
block in quick on fxp3 all head 70

# Backup - Allow all machines at Database/Applications segment
communicate with Backup server
pass in quick on fxp3 proto tcp from 192.168.3.0/24 port >
1023 to 192.168.1.9/32 port = 6101 flags S/SA keep state group
70

# access the DNS cache/resolver - Allow all machines at
Database/Applications segment communicate with the DNS
cache/resolver at Management/Services segment
pass in quick on fxp3 proto udp from 192.168.3.0/24 port >
1023 to 192.16.8.1.8/32 port = 53 keep state group 70

# send logs to Syslog - Allow all machines at
Database/Applications segment send their logs to Loghost
(goiaba)
pass in quick on fxp3 proto udp from 192.168.3.0/24 port >
1023 to 192.16.8.1.3/32 port = 514 keep state group 70

# synchronize the system clock with the local NTP server
pass in quick on fxp3 proto udp from 192.168.3.0/24 port = 123
to 192.16.8.1.1/32 port = 123 keep state group 70

# get updates from RedHat Network (xmlrpc.rhn.redhat.com)
pass in quick on fxp3 proto tcp from 192.168.3.0/24 port >
1023 to 66.187.232.101/32 port = 443 flags S/SA keep state
group 70

# block everything else
block in log quick on fxp3 all group 70

# fxp3 OUT (somewhere -> database/applications)
block out quick on fxp3 all head 80

# accepts SSH connections from Services-Management segment
pass out quick on fxp3 proto tcp from 192.168.1.0/24 port >
1023 to 192.168.3.8/32 port = 22 flags S/SA keep state group
80
pass out quick on fxp3 proto tcp from 192.168.1.0/24 port >
1023 to 192.168.3.9/32 port = 22 flags S/SA keep state group
80

# accept packets from web cluster servers to Jboss application
server
```

```
pass out quick on fxp3 proto tcp from 172.16.1.33/32 port >
1023 to 192.168.3.8/32 port = 8443 flags S/SA keep state group
80
pass out quick on fxp3 proto tcp from 172.16.1.34/32 port >
1023 to 192.168.3.8/32 port = 8443 flags S/SA keep state group
80
pass out quick on fxp3 proto tcp from 172.16.1.35/32 port >
1023 to 192.168.3.8/32 port = 8443 flags S/SA keep state group
80
pass out quick on fxp3 proto tcp from 172.16.1.36/32 port >
1023 to 192.168.3.8/32 port = 8443 flags S/SA keep state group
80
pass out quick on fxp3 proto tcp from 172.16.1.33/32 port >
1023 to 192.168.3.9/32 port = 8443 flags S/SA keep state group
80
pass out quick on fxp3 proto tcp from 172.16.1.34/32 port >
1023 to 192.168.3.9/32 port = 8443 flags S/SA keep state group
80
pass out quick on fxp3 proto tcp from 172.16.1.35/32 port >
1023 to 192.168.3.9/32 port = 8443 flags S/SA keep state group
80
pass out quick on fxp3 proto tcp from 172.16.1.36/32 port >
1023 to 192.168.3.9/32 port = 8443 flags S/SA keep state group
80

# accept packets from cocada, the Backup server
pass out quick on fxp3 proto tcp from 192.168.1.9/32 port >
1023 to 192.168.3.0/24 port 8191 >< 8194 flags S/SA keep state
group 80
# accept packets from tapioca SNMP management station
pass out quick on fxp3 proto udp from 192.168.1.10/32 port >
1023 to 192.168.3.4/32 port = 161 keep state group 80
pass out quick on fxp3 proto udp from 192.168.1.10/32 port >
1023 to 192.168.3.5/32 port = 161 keep state group 80
pass out quick on fxp3 proto udp from 192.168.1.10/32 port >
1023 to 192.168.3.8/32 port = 161 keep state group 80
pass out quick on fxp3 proto udp from 192.168.1.10/32 port >
1023 to 192.168.3.9/32 port = 161 keep state group 80

# block everything else
block out log quick on fxp3 all group 80

# fxp4 IN (IDS segment -> somewhere)
block in quick on fxp4 all head 90

# access the DNS cache/resolver
pass in quick on fxp4 proto udp from 192.168.255.0/24 port >
1023 to 192.16.8.1.8/32 port = 53 keep state group 90

# Allow all machines at IDS segment to synchronize the system
clock with the local NTP server
```

```
pass in quick on fxp4 proto udp from 192.168.255.0/24 port =
123 to 192.16.8.1.1/32 port = 123 keep state group 90

# Allow all machines access at IDS segment the snort.org
website to get rules and program updates
pass in quick on fxp4 proto tcp from 192.168.255.0/24 port >
1023 to 199.107.65.177/32 port = 80 flags S/SA keep state
group 90

# Allow all machines at IDS segment to get updates from RedHat
Network (xmlrpc.rhn.redhat.com)
pass in quick on fxp3 proto tcp from 192.168.255.0/24 port >
1023 to 66.187.232.101/32 port = 443 flags S/SA keep state
group 90

# block everything else
block in quick on fxp4 all group 90

# fxp4 OUT (somewhere -> IDS segment)
block out quick on fxp4 all head 100

# This segment does not offer services
block in log quick all group 100
```

2.4 - VPN server

The VPN gateway allow teleworkers and salespeople to connect to GIAC's network securely using IPsec and digital certificates. A Red Hat GNU/Linux 9.0 with FreeS/WAN is used to provide a secure communication channel between the remote user and GIAC's network.

2.4.1 - General system configuration

An unmodified kernel is installed in the system before get patched by FreeS/WAN installation. The kernel version used in this paper is 2.4.22. FreeS/WAN and Kernel installation isn't discussed in this paper.

2.4.2 - X509 digital certificates

Using digital certificates with IPSec is easiar than pre-shared keys, because it's not essentially secure if you have too many people knowing the secret. Althought issuing X509 certificates isn't a good idea if you need many people to trust you. In this cenario, create a Root CA and issue certificates will become very difficult and very expensive. In GIAC's case, only teleworkers and salespeople need to trust the certificates, and will not use these certificates to do any kind of business. If a certificate becomes comprimised, you just need to revoke and issue another one.

GIAC Enterprises decided to create a Root CA and issue certificates using OpenSSL, a free tool that can be used to issue certificates.

2.4.2.1 - Creating a Root CA with OpenSSL

First it is necessary to make a few modifications on /usr/share/ssl/openssl.cnf on Red Hat GNU/Linux before creating a auto-signed CA certificate.

In the /usr/share/ssl/misc/openssl.cnf file, it is necessary to modify the ``default_days" value to 3650 (ten years) and ``default_bits" value to 2048 (bits). The Root CA certificate should be stronger and be valid for a long period of time. The command used to create the certificate is shown below:

```
# cd /usr/share/ssl/misc
# ./CA -newca
CA certificate filename (or enter to create)

Making CA certificate ...
Generating a 2048 bit RSA private key
..........+++
...............+++
writing new private key to './demoCA/private/./cakey.pem'
Enter PEM pass phrase: (enter the passphrase)
Verifying - Enter PEM pass phrase: (enter the same passphrase)
-----
You are about to be asked to enter information that will be
incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished
Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [BR]:BR
State or Province Name (full name) [Sao Paulo]:Sao Paulo
Locality Name (eg, city) [Campinas]:Campinas
Organization Name (eg, company) [GIAC Enterprises]:GIAC
Enterprises
Organizational Unit Name (eg, section) []: (left blank)
Common Name (eg, your name or your server's hostname)
[]:ca.giacfortunecookies.com
Email Address []:ca@giacfortunecookies.com
```

This certificate will be used to sign the VPN gateway and RoadWarriors certificates. The password used must be stored in a secure place for further usage. At this time it is a good idea to revert all modifications made on /usr/share/ssl/openssl.cnf configuration file.

The CA certificate need to be copied to /etc/ipsec.d/cacerts directory, in order to be used by FreeS/WAN.

```
# cd /usr/share/ssl/misc
# cp demoCA/cacert.pem /etc/ipsec.d/cacerts/myCAcert.pem
```

2.4.2.2 - Creating the Certificate Revocation List (CRL)

To create and install a CRL for 30 days, a sequence of commands must be done. The password used to create the CA certificate is needed in this process.

```
# cd /usr/share/ssl/misc
# openssl ca -gencrl -crldays 30 -out
/etc/ipsec.d/crls/myCrl.pem
Using configuration form /usr/share/ssl/openssl.cnf
Enter PEM pass phrase:
```

2.4.2.3 - Creating the certificate for the VPN gateway

To create the certificate used by the VPN gateway, it is necessary to make one modification on /usr/share/ssl/openssl.cnf. Just add the the entry ``subjectAltName=DNS:copy" in ``[usr_cert]" section. The Fully Qualified Domain Name (FQDN) id will be used during the authentication process. The command used to create the certificate is shown below:

```
# cd /usr/share/ssl/misc
# ./CA -newcert
Generating a 1024 bit RSA private key
.........................................++++++
.....++++++
writing new private key to 'newreq.pem'
Enter PEM pass phrase: (enter the passphrase)
Verifying - Enter PEM pass phrase: (enter the passphrase)
-----
You are about to be asked to enter information that will be
incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished
Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [BR]:BR
State or Province Name (full name) [Sao Paulo]:Sao Paulo
Locality Name (eg, city) [Campinas]:Campinas
Organization Name (eg, company) [GIAC Enterprises]:GIAC
Enterprises
Organizational Unit Name (eg, section) []:
Common Name (eg, your name or your server's hostname)
[]:rapadura.giacfortunecookies.com
Email Address []:certs@giacfortunecookies.com
Certificate (and private key) is in newreq.pem
```

Now it is necessary to sign the certificate request with the Root CA certificate. The CA's and certificate request password are required in this process.

```
# cd /usr/share/ssl/misc
# ./CA -signcert
Cert passphrase will be requested twice - bug?
Getting request Private Key
Enter pass phrase for newreq.pem:
Generating certificate request
Using configuration from /usr/share/ssl/openssl.cnf
```

```
Enter pass phrase for ./demoCA/private/cakey.pem:
Check that the request matches the signature
Signature ok
Certificate Details:
Serial Number: 1 (0x1)
Validity
    Not Before: Sep 19 00:15:07 2003 GMT
    Not After : Sep 18 00:15:07 2004 GMT
Subject:
    countryName               = BR
    stateOrProvinceName       = Sao Paulo
    localityName              = Campinas
    organizationName          = GIAC Enterprises
    commonName                = 
rapadura.giacfortunecookies.com
    emailAddress              = certs@giacfortunecookies.com
X509v3 extensions:
    X509v3 Basic Constraints:
    CA:FALSE
    Netscape Comment:
    OpenSSL Generated Certificate
    X509v3 Subject Key Identifier:

B2:A7:7F:07:39:46:F3:EB:C0:0E:E9:47:99:F1:C2:FC:FB:4F:6E:72
    X509v3 Authority Key Identifier:

keyid:28:80:25:33:0D:0C:18:8F:02:10:44:70:04:35:D3:3C:9D:D4:89
:D5
    DirName:/C=BR/ST=Sao Paulo/L=Campinas/O=GIAC
Enterprises/CN=ca.giacfortunecookies.com/emailAddress=ca@giacf
ortunecookies.com
    serial:00

    X509v3 Subject Alternative Name:
    DNS:copy
Certificate is to be certified until Sep 18 00:15:07 2004 GMT
(365 days)
Sign the certificate? [y/n]:y (answer yes to accept)

1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
Certificate:
Data:
Version: 3 (0x2)
Serial Number: 1 (0x1)
Signature Algorithm: md5WithRSAEncryption
Issuer: C=BR, ST=Sao Paulo, L=Campinas, O=GIAC Enterprises,
CN=ca.giacfortunecookies.com/emailAddress=ca@giacfortunecookie
s.com
Validity
```

```
    Not Before: Sep 19 00:15:07 2003 GMT
    Not After : Sep 18 00:15:07 2004 GMT
Subject: C=BR, ST=Sao Paulo, L=Campinas, O=GIAC Enterprises,
CN=rapadura.giacfortunecookies.com/emailAddress=certs@giacfort
unecookies.com
Subject Public Key Info:
    Public Key Algorithm: rsaEncryption
    RSA Public Key: (1024 bit)
      Modulus (1024 bit):
          00:df:13:8e:83:6f:6a:51:63:dd:73:b0:91:75:ae:
          64:cc:cc:9b:6a:22:72:40:37:9d:13:a7:39:be:8e:
          cb:57:cc:5f:2d:01:83:4d:e1:1a:2e:42:2f:2c:84:
          e9:73:27:b8:8c:74:e6:f5:5b:96:d5:fe:55:66:5f:
          a5:24:6d:04:6f:e7:53:e7:46:4f:4d:2b:4a:c4:50:
          53:0b:80:af:f2:68:9a:eb:91:fc:40:1e:9e:0d:eb:
          b5:0b:bd:dd:8b:1f:84:de:60:5a:aa:06:bc:95:43:
          a1:20:65:ce:cf:64:8f:e0:1a:c1:f4:7d:6f:83:3c:
          7d:58:27:84:98:e5:67:30:c5
      Exponent: 65537 (0x10001)
X509v3 extensions:
    X509v3 Basic Constraints:
    CA:FALSE
    Netscape Comment:
    OpenSSL Generated Certificate
    X509v3 Subject Key Identifier:

B2:A7:7F:07:39:46:F3:EB:C0:0E:E9:47:99:F1:C2:FC:FB:4F:6E:72
    X509v3 Authority Key Identifier:

keyid:28:80:25:33:0D:0C:18:8F:02:10:44:70:04:35:D3:3C:9D:D4:89
:D5
    DirName:/C=BR/ST=Sao Paulo/L=Campinas/O=GIAC
Enterprises/CN=ca.giacfortunecookies.com/emailAddress=ca@giacf
ortunecookies.com
    serial:00

    X509v3 Subject Alternative Name:
    DNS:copy
Signature Algorithm: md5WithRSAEncryption
4b:fc:9e:ab:09:ec:4b:23:7c:87:fd:f5:8c:72:cb:a0:59:08:
b6:eb:ec:52:f6:79:42:24:47:af:5b:c7:e0:25:2b:59:ec:7b:
41:ae:d2:ee:0c:92:02:19:6c:ee:70:91:99:79:fb:ca:67:96:
b1:cf:cb:7a:14:61:7a:f8:ac:51:b7:eb:21:32:5a:a1:5c:3b:
67:34:36:10:60:ef:f4:3c:ee:91:cf:e0:26:48:81:45:a1:05:
cd:93:50:34:40:71:05:ac:9a:f2:d4:e7:e5:db:15:45:61:54:
20:33:f1:a9:3e:33:a0:a7:8c:fc:29:0a:1f:b3:60:2d:73:68:
cf:29:22:e3:de:66:de:cb:4f:b0:d9:11:a8:d0:da:99:cc:78:
b9:fb:35:41:54:7a:41:dd:9f:2e:0b:5d:63:5c:98:ad:12:99:
c1:cf:97:dc:25:34:34:1b:f1:d5:50:e6:40:d5:e1:ba:1f:97:
e5:b8:8c:5c:64:a4:94:f6:3e:e6:52:38:14:d9:f2:fa:05:94:
17:93:52:a6:72:d6:b7:48:53:f1:f3:db:98:fb:31:c5:b9:d9:
```

```
2a:83:43:41:71:ed:22:8c:45:43:16:84:e8:98:f0:25:6f:cc:
80:40:a2:c8:ec:08:a8:cf:2d:5d:c9:78:04:b9:1b:97:fc:7e:
88:ea:d9:9e
-----BEGIN CERTIFICATE-----
MIIEeDCCA2CgAwIBAgIBATANBgkqhkiG9w0BAQQFADCBnTELMAkGA1UEBhMCQl
Ix
EjAQBgNVBAgTCVNhbyBQYXVsbzERMA8GA1UEBxMIQ2FtcGluYXMxGTAXBgNVBA
oT
EEdJQUMgRW50ZXJwcmlzZXMxIjAgBgNVBAMTGWNhLmdpYWNmb3J0dW5lY29va2
ll
cy5jb20xKDAmBgkqhkiG9w0BCQEWGWNhQGdpYWNmb3J0dW5lY29va2llcy5jb2
0w
HhcNMDMwOTE5MDAxNTA3WhcNMDQwOTE4MDAxNTA3WjCBpjELMAkGA1UEBhMCQl
Ix
EjAQBgNVBAgTCVNhbyBQYXVsbzERMA8GA1UEBxMIQ2FtcGluYXMxGTAXBgNVBA
oT
EEdJQUMgRW50ZXJwcmlzZXMxKDAmBgNVBAMTH3JhcGFkdXJhLmdpYWNmb3J0dW
5l
Y29va2llcy5jb20xKzApBgkqhkiG9w0BCQEWHGNlcnRzQGdpYWNmb3J0dW5lY2
9v
a2llcy5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAN8TjoNvalFj3X
Ow
kXWuZMzMm2oickA3nROnOb6Oy1fMXy0Bg03hGi5CLyyE6XMnuIx05vVbltX+VW
Zf
pSRtBG/nU+dGT00rSsRQUwuAr/JomuuR/EAeng3rtQu93YsfhN5gWqoGvJVDoS
Bl
zs9kj+AawfR9b4M8fVgnhJjlZzDFAgMBAAGjggE6MIIBNjAJBgNVHRMEAjAAMC
wG
CWCGSAGG+EIBDQQfFh1PcGVuU1NMIEdlbmVyYXRlZCBDZXJ0aWZpY2F0ZTAdBg
NV
HQ4EFgQUsqd/BzlG8+vADulHmfHC/PtPbnIwgcoGA1UdIwSBwjCBv4AUKIAlMw
0M
GI8CEERwBDXTPJ3UidWhgaOkgaAwgZ0xCzAJBgNVBAYTAkJSMRIwEAYDVQQIEw
lT
YW8gUGF1bG8xETAPBgNVBAcTCENhbXBpbmFzMRkwFwYDVQQKExBHSUFDIEVudG
Vy
cHJpc2VzMSIwIAYDVQQDExljYS5naWFjZm9ydHVuZWNvb2tpZXMuY29tMSgwJg
YJ
KoZIhvcNAQkBFhljYUBnaWFjZm9ydHVuZWNvb2tpZXMuY29tggEAMA8GA1UdEQ
QI
MAaCBGNvcHkwDQYJKoZIhvcNAQEEBQADggEBAEv8nqsJ7EsjfIf99Yxyy6BZCL
br
7FL2eUIkR69bx+AlK1nse0Gu0u4MkgIZbO5wkZl5+8pnlrHPy3oUYXr4rFG36y
Ey
WqFcO2c0NhBg7/Q87pHP4CZIgUWhBc2TUDRAcQWsmvLU5+XbFUVhVCAz8ak+M6
Cn
jPwpCh+zYC1zaM8pIuPeZt7LT7DZEajQ2pnMeLn7NUFUekHdny4LXWNcmK0Smc
HP
l9wlNDQb8dVQ5kDV4bofl+W4jFxkpJT2PuZSOBTZ8voFlBeTUqZy1rdIU/Hz25
j7
McW52SqDQ0Fx7SKMRUMWhOiY8CVvzIBAosjsCKjPLV3JeAS5G5f8fojq2Z4=
-----END CERTIFICATE-----
Signed certificate is in newcert.pem
```

The certificate and private key must be copied the correct directories, /etc/ipsec.d/certs and /etc/ipsec.d/private, respectively.

```
# cd /usr/share/ssl/misc
# cp -p newcert.pem /etc/ipsec.d/certs/myCert.pem
# cp -p newreq.pem /etc/ipsec.d/private/myKey.pem
```

2.4.2.4 - Creating RoadWarriors certificates

To create the RoadWarriors certificates, again, it is necessary to make one modification on /usr/share/ssl/misc/openssl.cnf configuration file. The entry ``subjectAltName=DNS:copy" must be replaced with "subjectAltName=email:copy". The same procedure to create the certificate for the VPN gateway is used.

```
# cd /usr/share/ssl/misc
# ./CA -newcert
Generating a 1024 bit RSA private key
.............................................................
.........++++++
...........++++++
writing new private key to 'newreq.pem'
Enter PEM pass phrase: (enter the passphrase)
Verifying - Enter PEM pass phrase: (enter the passphrase)
-----
You are about to be asked to enter information that will be
incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished
Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [BR]:
State or Province Name (full name) [Sao Paulo]:
Locality Name (eg, city) [Campinas]:
Organization Name (eg, company) [GIAC Enterprises]:
Organizational Unit Name (eg, section) []:
Common Name (eg, your name or your server's hostname)
[]:roadwarrior@giacfortunecookies.com
Email Address []:roadwarrior@giacfortunecookies.com
Certificate (and private key) is in newreq.pem
```

Again, it is necessary to sign the certificate requet with the Root CA certificate. The CA's and certificate request passwords are required in this process. All RoadWarriors certificates are created using the proccess described above.

```
# ./CA -signcert
Cert passphrase will be requested twice - bug?
Getting request Private Key
Enter pass phrase for newreq.pem:
Generating certificate request
```

```
Using configuration from /usr/share/ssl/openssl.cnf
Enter pass phrase for ./demoCA/private/cakey.pem:
Check that the request matches the signature
Signature ok
Certificate Details:
Serial Number: 2 (0x2)
Validity
    Not Before: Sep 19 00:39:01 2003 GMT
    Not After : Sep 18 00:39:01 2004 GMT
Subject:
    countryName               = BR
    stateOrProvinceName       = Sao Paulo
    localityName              = Campinas
    organizationName          = GIAC Enterprises
    commonName                =
roadwarrior@giacfortunecookies.com
    emailAddress              =
roadwarrior@giacfortunecookies.com
X509v3 extensions:
    X509v3 Basic Constraints:
    CA:FALSE
    Netscape Comment:
    OpenSSL Generated Certificate
    X509v3 Subject Key Identifier:

01:38:74:8F:E8:6A:AB:9F:91:EA:0E:8C:25:89:3E:A0:AA:E9:DA:65
    X509v3 Authority Key Identifier:

keyid:28:80:25:33:0D:0C:18:8F:02:10:44:70:04:35:D3:3C:9D:D4:89
:D5
    DirName:/C=BR/ST=Sao Paulo/L=Campinas/O=GIAC
Enterprises/CN=ca.giacfortunecookies.com/emailAddress=ca@giacf
ortunecookies.com
    serial:00

    X509v3 Subject Alternative Name:
    email:roadwarrior@giacfortunecookies.com
Certificate is to be certified until Sep 18 00:39:01 2004 GMT
(365 days)
Sign the certificate? [y/n]:y

1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
Certificate:
Data:
Version: 3 (0x2)
Serial Number: 2 (0x2)
Signature Algorithm: md5WithRSAEncryption
```

```
Issuer: C=BR, ST=Sao Paulo, L=Campinas, O=GIAC Enterprises,
CN=ca.giacfortunecookies.com/emailAddress=ca@giacfortunecookie
s.com
Validity
    Not Before: Sep 19 00:39:01 2003 GMT
    Not After : Sep 18 00:39:01 2004 GMT
Subject: C=BR, ST=Sao Paulo, L=Campinas, O=GIAC Enterprises,
CN=roadwarrior@giacfortunecookies.com/emailAddress=roadwarrior
@giacfortunecookies.com
Subject Public Key Info:
    Public Key Algorithm: rsaEncryption
    RSA Public Key: (1024 bit)
      Modulus (1024 bit):
          00:9e:42:f0:5d:4b:d8:4c:2f:ac:20:b4:eb:fb:36:
          9e:2a:6f:1f:e1:8c:4a:58:fe:e8:59:7d:38:d4:f1:
          eb:0b:32:81:6e:d3:a7:32:00:6f:71:7a:4c:bd:b4:
          2f:d4:84:86:6e:47:c2:d4:f5:cb:c0:54:9c:6f:15:
          24:6e:1c:4d:8a:48:bb:ff:52:2b:26:7e:26:ac:79:
          d0:f4:2c:f1:a1:45:e3:1c:bc:9e:3a:37:15:40:8a:
          38:9d:f9:76:8d:4d:25:35:ec:e3:68:8c:dd:4f:80:
          a3:d3:e8:b7:f8:43:2e:17:74:c4:dc:a4:38:95:15:
          a3:f4:7d:b6:f6:b2:22:5f:8d
      Exponent: 65537 (0x10001)
X509v3 extensions:
    X509v3 Basic Constraints:
    CA:FALSE
    Netscape Comment:
    OpenSSL Generated Certificate
    X509v3 Subject Key Identifier:

01:38:74:8F:E8:6A:AB:9F:91:EA:0E:8C:25:89:3E:A0:AA:E9:DA:65
    X509v3 Authority Key Identifier:

keyid:28:80:25:33:0D:0C:18:8F:02:10:44:70:04:35:D3:3C:9D:D4:89
:D5
    DirName:/C=BR/ST=Sao Paulo/L=Campinas/O=GIAC
Enterprises/CN=ca.giacfortunecookies.com/emailAddress=ca@giacf
ortunecookies.com
    serial:00

    X509v3 Subject Alternative Name:
    email:roadwarrior@giacfortunecookies.com
Signature Algorithm: md5WithRSAEncryption
45:d0:0a:0b:3f:b0:25:80:05:7b:c8:d1:d0:af:6c:3f:2b:4e:
2e:e5:e1:8d:26:e1:9e:e5:fe:2b:77:a0:8b:ee:00:e0:3e:fb:
c2:4c:de:3f:f0:b2:2a:12:b2:24:52:16:05:80:a0:1c:72:8c:
f4:51:51:7f:c2:72:5b:e5:df:bf:7e:cb:a4:67:4e:f1:a4:80:
67:14:d7:1f:35:86:b6:5b:0c:3a:6b:1a:45:3b:6f:ca:d5:17:
e2:79:75:a5:a7:5c:e4:50:01:47:9c:83:c1:4b:ff:2d:db:bc:
3e:e6:9b:04:00:35:4b:a0:72:6f:65:f1:b9:75:3b:b6:42:e5:
2b:4a:e5:85:0e:2e:3d:c4:74:64:6f:0b:7f:0d:db:d8:95:4a:
```

```
bd:b9:8b:d3:0b:0f:42:06:b6:4c:b6:34:a3:8b:0d:64:57:07:
35:25:f0:ca:09:10:fa:92:b2:3a:46:ff:85:6d:a3:03:ba:2a:
3e:38:80:0e:63:b7:da:e5:d6:51:a2:a0:9c:d0:36:b7:c7:d9:
41:6a:cf:f6:80:4e:65:35:88:28:b3:f4:e1:99:97:b7:1b:09:
81:4b:2b:21:9a:c4:14:a2:6d:0b:29:77:89:46:22:88:b4:1c:
d3:3d:eb:5d:4b:40:09:db:71:db:22:65:3e:21:ab:31:74:50:
c6:ea:2c:e6
-----BEGIN CERTIFICATE-----
MIIEnzCCA4egAwIBAgIBAjANBgkqhkiG9w0BAQQFADCBnTELMAkGA1UEBhMCQl
Ix
EjAQBgNVBAgTCVNhbyBQYXVsbzERMA8GA1UEBxMIQ2FtcGluYXMxGTAXBgNVBA
oT
EEdJQUMgRW50ZXJwcmlzZXMxIjAgBgNVBAMTGWNhLmdpYWNmb3J0dW5lY29va2
ll
cy5jb20xKDAmBgkqhkiG9w0BCQEWGWNhQGdpYWNmb3J0dW5lY29va2llcy5jb2
0w
HhcNMDMwOTE5MDAzOTAxWhcNMDQwOTE4MDAzOTAxWjCBrzELMAkGA1UEBhMCQl
Ix
EjAQBgNVBAgTCVNhbyBQYXVsbzERMA8GA1UEBxMIQ2FtcGluYXMxGTAXBgNVBA
oT
EEdJQUMgRW50ZXJwcmlzZXMxKzApBgNVBAMUInJvYWR3YXJyaW9yQGdpYWNmb3
J0
dW5lY29va2llcy5jb20xMTAvBgkqhkiG9w0BCQEWInJvYWR3YXJyaW9yQGdpYW
Nm
b3J0dW5lY29va2llcy5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAJ
5C
8F1L2EwvrCC06/s2nipvH+GMSlj+6Fl9ONTx6wsygW7TpzIAb3F6TL20L9SEhm
5H
wtT1y8BUnG8VJG4cTYpIu/9SKyZ+Jqx50PQs8aFF4xy8njo3FUCKOJ35do1NJT
Xs
42iM3U+Ao9Pot/hDLhd0xNykOJUVo/R9tvayIl+NAgMBAAGjggFYMIIBVDAJBg
NV
HRMEAjAAMCwGCWCGSAGG+EIBDQQfFh1PcGVuU1NMIEdlbmVyYXRlZCBDZXJ0aW
Zp
Y2F0ZTAdBgNVHQ4EFgQUATh0j+hqq5+R6g6MJYk+oKrp2mUwgcoGA1UdIwSBwj
CB
v4AUKIAlMw0MGI8CEERwBDXTPJ3UidWhgaOkgaAwgZ0xCzAJBgNVBAYTAkJSMR
Iw
EAYDVQQIEwlTYW8gUGF1bG8xETAPBgNVBAcTCENhbXBpbmFzMRkwFwYDVQQKEx
BH
SUFDIEVudGVycHJpc2VzMSIwIAYDVQQDExljYS5naWFjZm9ydHVuZWNvb2tpZX
Mu
Y29tMSgwJgYJKoZIhvcNAQkBFhljYUBnaWFjZm9ydHVuZWNvb2tpZXMuY29tgg
EA
MC0GA1UdEQQmMCSBInJvYWR3YXJyaW9yQGdpYWNmb3J0dW5lY29va2llcy5jb2
0w
DQYJKoZIhvcNAQEEBQADggEBAEXQCgs/sCWABXvI0dCvbD8rTi7l4Y0m4Z7l/i
t3
oIvuAOA++8JM3j/wsioSsiRSFgWAoBxyjPRRUX/Cclvl379+y6RnTvGkgGcU1x
81
```

```
hrZbDDprGkU7b8rVF+J5daWnXORQAUecg8FL/y3bvD7mmwQANUugcm9l8bl1O7
ZC
5StK5YUOLj3EdGRvC38N29iVSr25i9MLD0IGtky2NKOLDWRXBzUl8MoJEPqSsj
pG
/4VtowO6Kj44gA5jt9rl1lGioJzQNrfH2UFqz/aATmU1iCiz9OGZl7cbCYFLKy
Ga
xBSibQspd4lGIoi0HNM9611LQAnbcdsiZT4hqzF0UMbqLOY=
-----END CERTIFICATE-----
Signed certificate is in newcert.pem
```

After that, the certificate is copied to the /etc/ipsec.d/certs directory, as follows:

```
# cd /usr/share/ssl/misc
# cp newcert.pem /etc/ipsec.d/certs/roadwarrior.pem
```

The RoadWarrior certificate must converted to the PKCS12 export format, in order to be imported by the SSH Sentinel IPsec client. The command used to convert the certificate is described below.

```
# cd /usr/share/ssl/misc
# openssl pkcs12 -export -inkey newreq.pem -in newcert.pem -
name "OpenSSL client certificate" \
-certfile demoCA/cacert.pem -caname "OpenSSL Root CA
certificate" -out roadwarrior_cert.p12
Enter PEM pass phrase: (use the passphrase used to create the
certificate)
Enter Export Password: (enter a passphrase used to export the
certificate)
Verifying password - Enter Export Password:
```

The certificate is password-protected but, a secure method is needed to deliver the certificate to its owner. In this case, the certificate is copied to a floppy disk that will be used in the IPsec client configuration process.

2.4.3 - FreeS/WAN configuration

In recent versions of the X509 patch for FreeS/WAN it is not necessary to convert the certificate to the DER format, so it is possible to configure the FreeS/WAN to read the certificate, just providing its password in /etc/ipsec.secrets.

```
: RSA myKey.pem "passphrase" # the passphrase used to create
the VPN gateway certificate
```

Only the root user should be able to read this file. It is a good idea to change the permissions of the /etc/ipsec.secrets file to 0600.

```
# chmod 600 /etc/ipsec.secrets
```

The /etc/ipsec.conf configuration file must contain information on how IPsec tunnels are created. In this paper, only one RoadWarrior is used, but is just a matter of creating new certificates and replicating parts of the configuration entries used in this example.

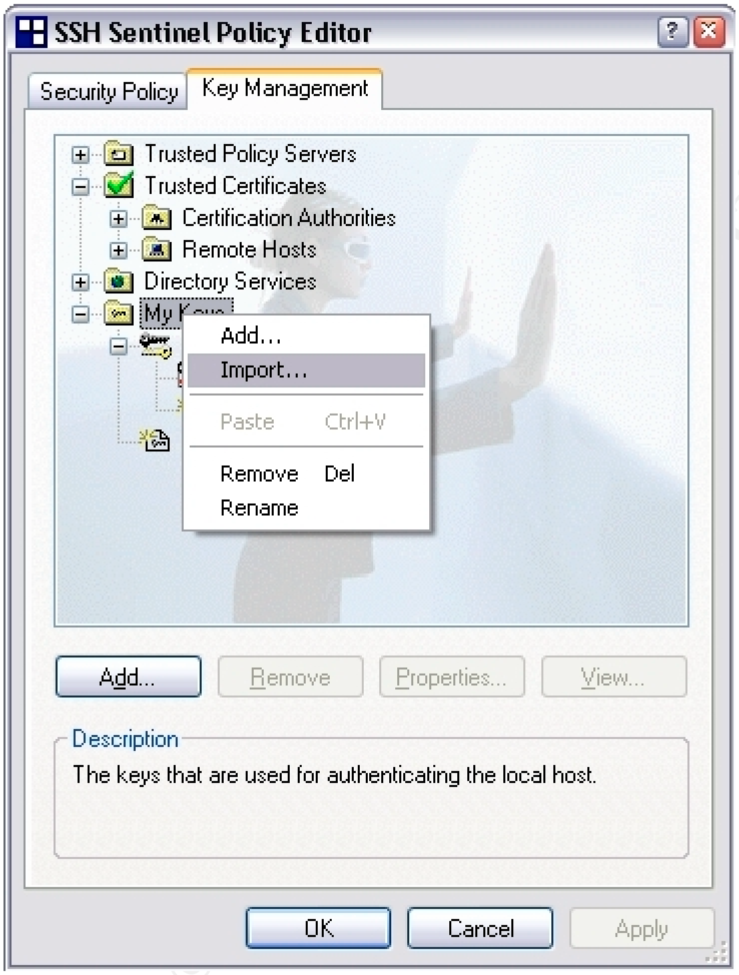
```
version 2.0

config setup
```

```
interfaces="ipsec0=eth0"
klipsdebug=none
plutodebug=none
#plutoload=%search   # not used in FreeS/WAN 2.x versions
#plutostart=%search  # not used in FreeS/WAN 2.x versions
uniqueids=yes
strictcrlpolicy=yes  # GIAC's Root CA issues Certificate
Revocation Lists (CRLs)

conn %default
keyingtries=3
disablearrivalcheck=no
authby=rsasig
keyexchange=ike
ikelifetime=240m
keylife=60m
compress=no
right=%any
rightrsasigkey=%cert
pfs=yes
left=172.16.1.253
leftnexthop=172.16.1.254
leftsubnet=192.168.2.0/24
leftid=@rapadura.giacfortunecookies.com
leftcert=myCert.pem
auth=esp
auto=add

conn block
auto=ignore

conn private
auto=ignore

conn private-or-clear
auto=ignore

conn clear-or-private
auto=ignore

conn clear
auto=ignore

conn roadwarrior-sentinel  # configuration entry for the
roadwarrior user.
type=tunnel
pfs=yes
right=%any
rightrsasigkey=%cert
rightcert=roadwarrior.pem
rightsubnetwithin=0.0.0.0/0
```

```
leftupdown=/usr/local/lib/ipsec/_updown_x509 # activate the
firewall rules for this particular connection
rightid=roadwarrior@cais.rnp.br
auto=add
```

2.4.4 - SSH Sentinel configuration

The details on installation procedure of SSH Sentinel isn't scope of this document, all necessary information is available on SSH website - www.ssh.com. An example of RoadWarrior certificate installation and configuration is described as follows. It is important to make sure the client is configured correctly, otherwise the connection will fail.



The keys that are used for authenticating the local host.

Figure 2 - Importing certificate

To import the RoadWarior certificate, right-click the SSH Sentinel icon in task bar, select ``Run Policy Editor'', click on ``Key Management'' tab, right-click on ``My Keys'' and select ``Import...''. Enter the export password and accept the certificate installation.

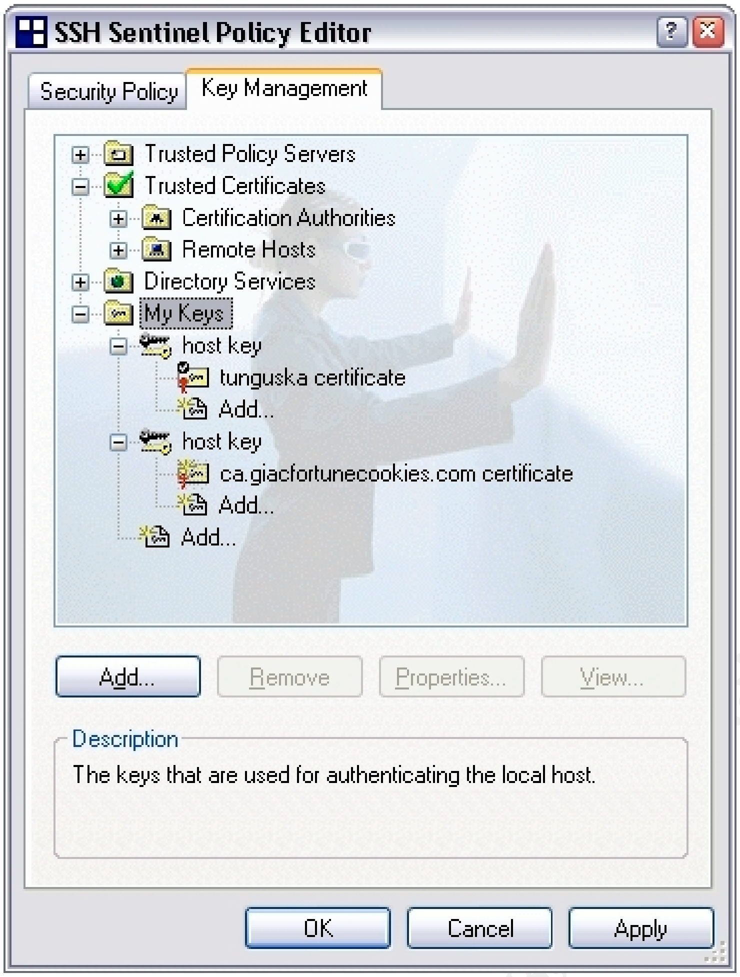
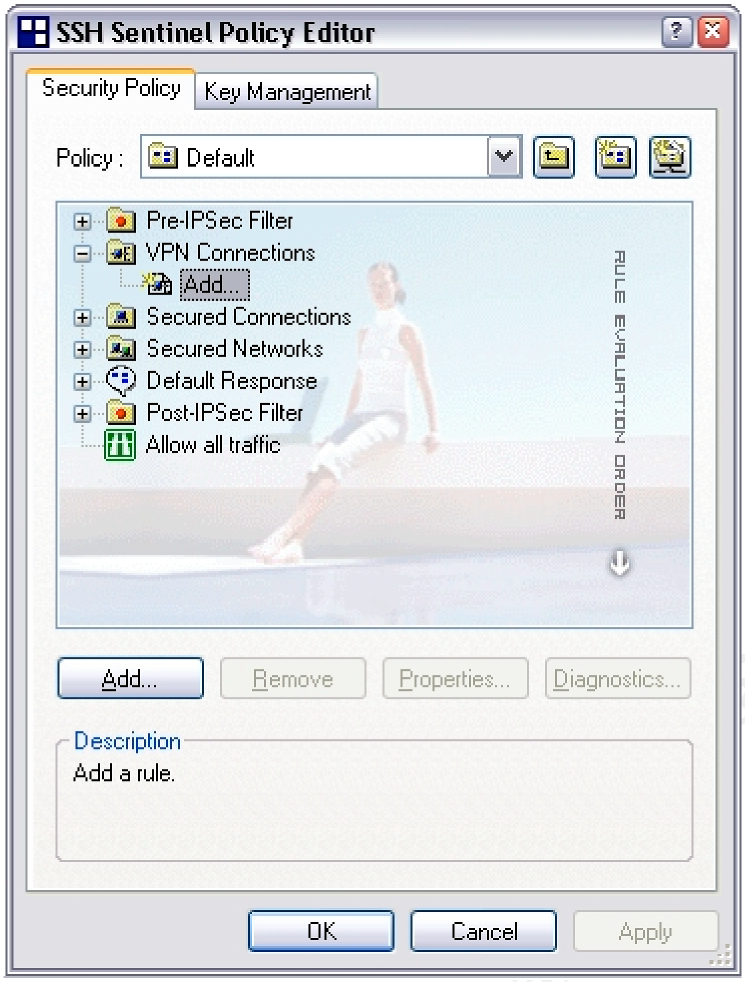


Figure 3 - RoadWarrior certificate installed

The RoadWarrior certificate is now installed as ``Host Key'' ca.giacfortunecookies.com certificate. It is necessary to click on apply to accept the configuration.

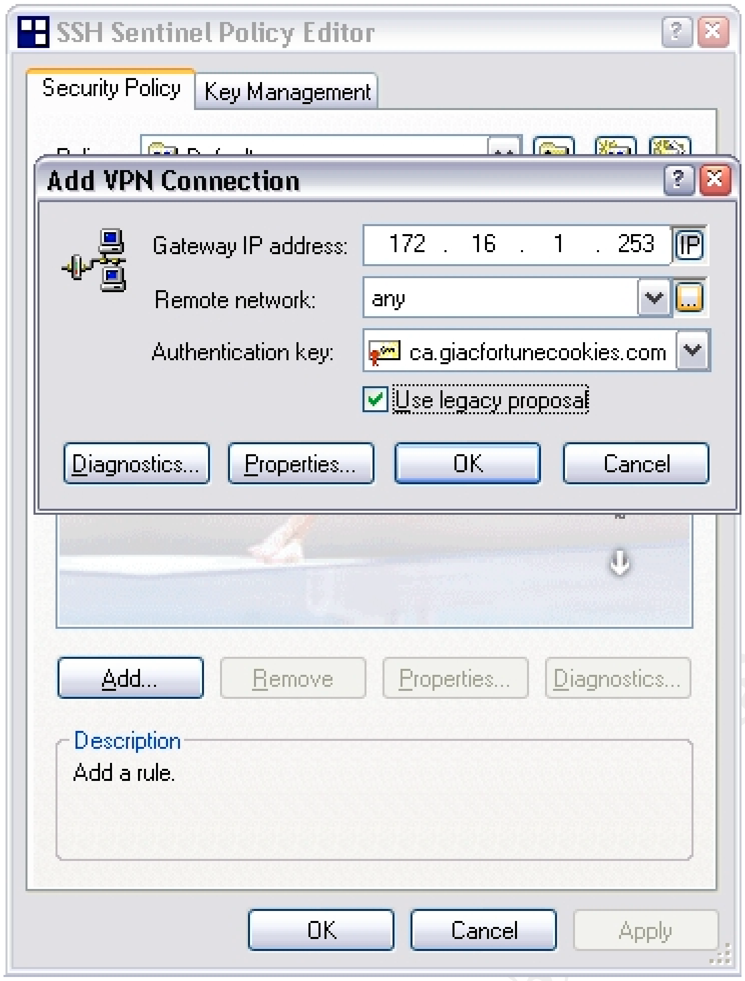


Policy : Default

Description
Add a rule.

Figure 4 - Creating the VPN connection

To create the VPN connection with GIAC Enterprise, click on the left tab ``Security Policy'' then click on ``Add'' button.



Gateway IP address: 172 . 16 . 1 . 253

Remote network: any

Authentication key: ca.giacfortunecookies.com

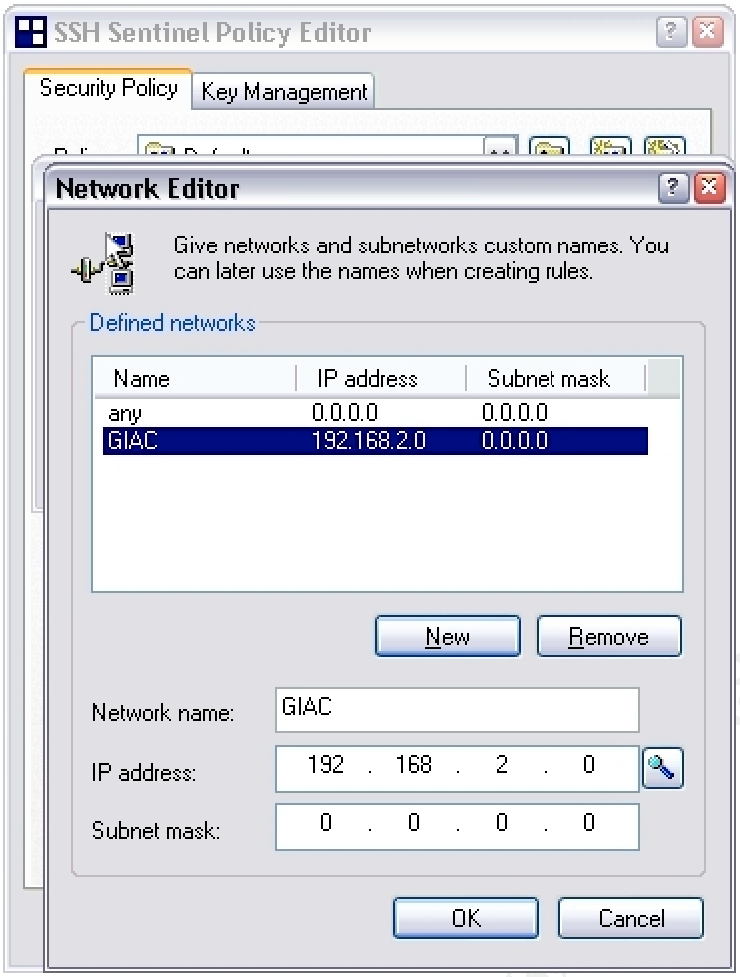
☑ Use legacy proposal

Description

Add a rule.

Figure 5 - Configuration parameters

Click the button ``IP" and enter the VPN gateway IP address (172.16.1.253) or enter the gateway's name (rapadura.giacfortunecookies.com). Select the ``Authentication Key" ca.giacfortunecookies.com, click on ``Use Legacy Proposal" then click on ``...".



Give networks and subnetworks custom names. You can later use the names when creating rules.

Defined networks

Name	IP address	Subnet mask
any	0.0.0.0	0.0.0.0
GIAC	192.168.2.0	0.0.0.0

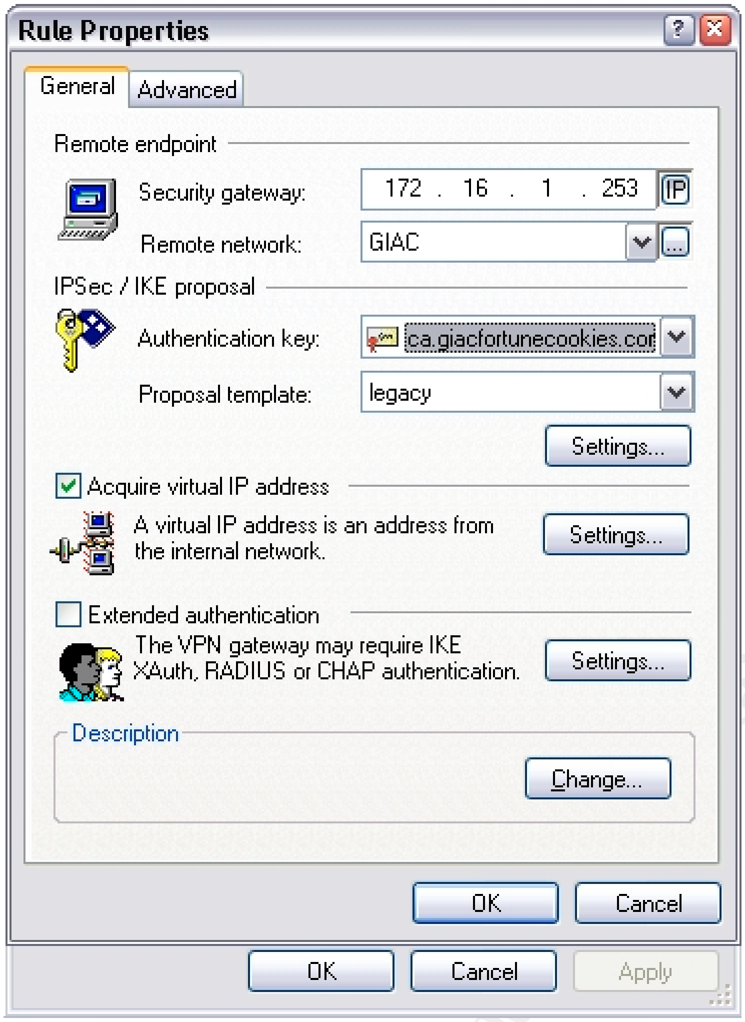
Network name: GIAC

IP address: 192 . 168 . 2 . 0

Subnet mask: 0 . 0 . 0 . 0

Figure 6 - Define the remote network and name

Click on ``New'' then enter the network name (GIAC, in this case), IP address and subnet mask. Click ``OK''.



Remote endpoint

Security gateway: 172 . 16 . 1 . 253 IP

Remote network: GIAC

IPSec / IKE proposal

Authentication key: ca.giacfortunecookies.cor

Proposal template: legacy

☑ Acquire virtual IP address

A virtual IP address is an address from the internal network.

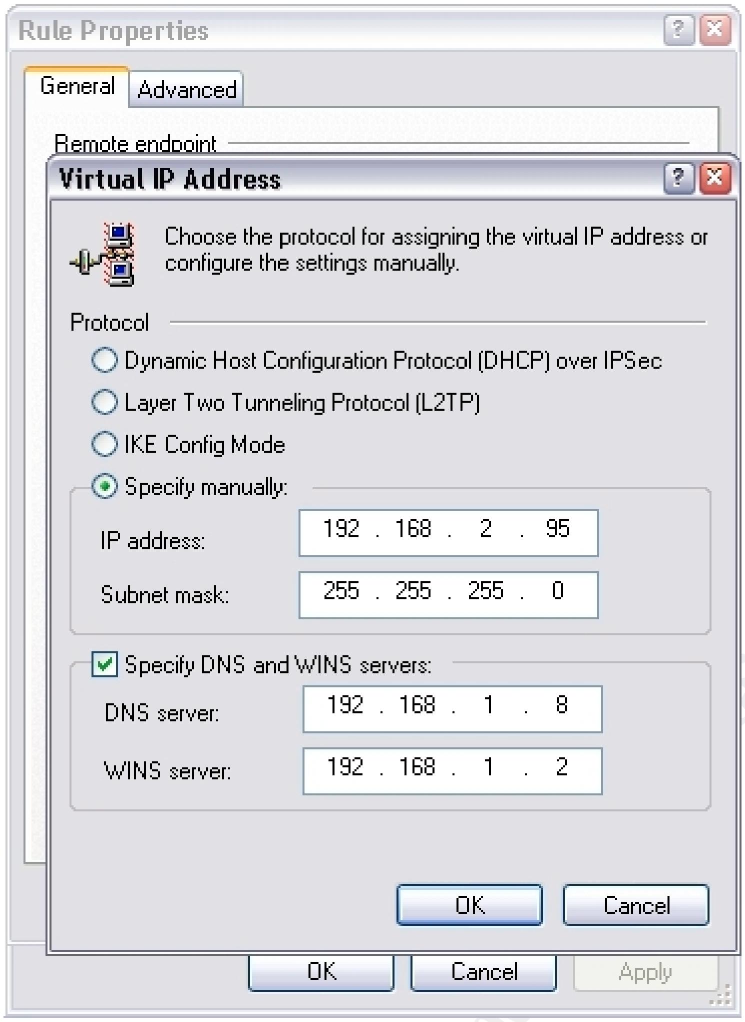
□ Extended authentication

The VPN gateway may require IKE XAuth, RADIUS or CHAP authentication.

Description

Figure 7 - Acquire Virtual IP Address

Select ``GIAC" on ``Remote Network" box, check the ``Acquire virtual IP address" box then click on ``Settings" button.



Choose the protocol for assigning the virtual IP address or configure the settings manually.

Protocol

○ Dynamic Host Configuration Protocol (DHCP) over IPSec
○ Layer Two Tunneling Protocol (L2TP)
○ IKE Config Mode
● Specify manually:

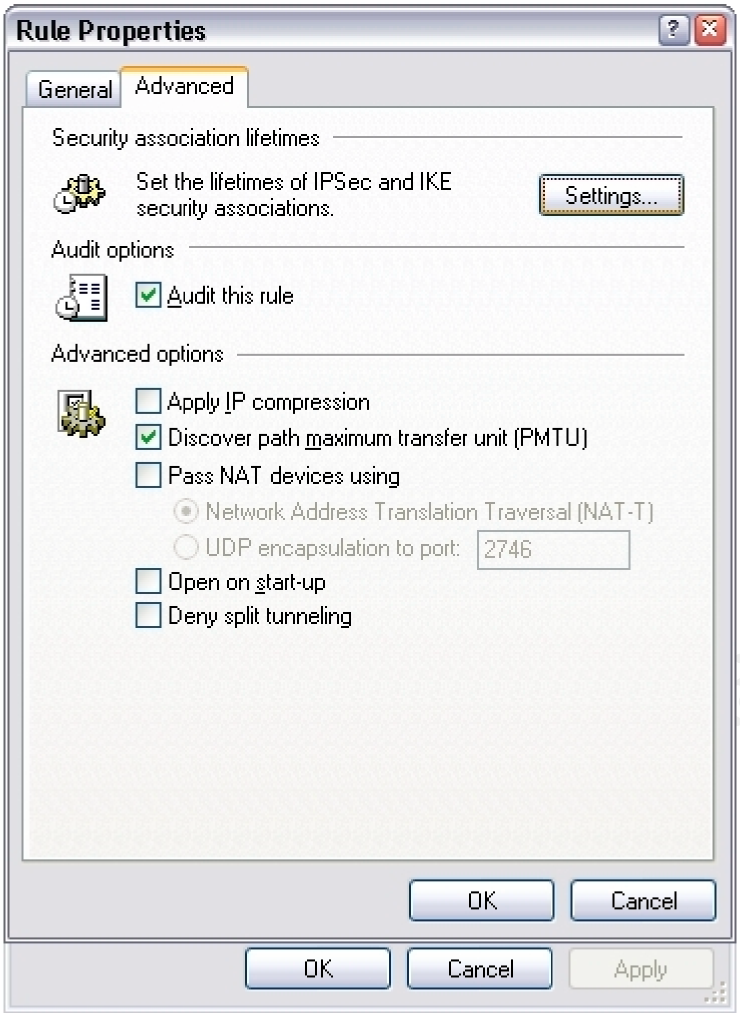
IP address: 192 . 168 . 2 . 95
Subnet mask: 255 . 255 . 255 . 0

☑ Specify DNS and WINS servers:

DNS server: 192 . 168 . 1 . 8
WINS server: 192 . 168 . 1 . 2

Figure 8 - Virtual IP Address definition

Select ``Specify Manually", enter the IP address and subnet mask. Select ``Specify DNS and WINS servers", enter the DNS and WINS server IP addresses. The virtual IP address is set manually and must not conflict with other machines. Click ``OK".



Security association lifetimes

Set the lifetimes of IPSec and IKE security associations.

Audit options

☑ Audit this rule

Advanced options

☐ Apply IP compression

☑ Discover path maximum transfer unit (PMTU)

☐ Pass NAT devices using

● Network Address Translation Traversal (NAT-T)

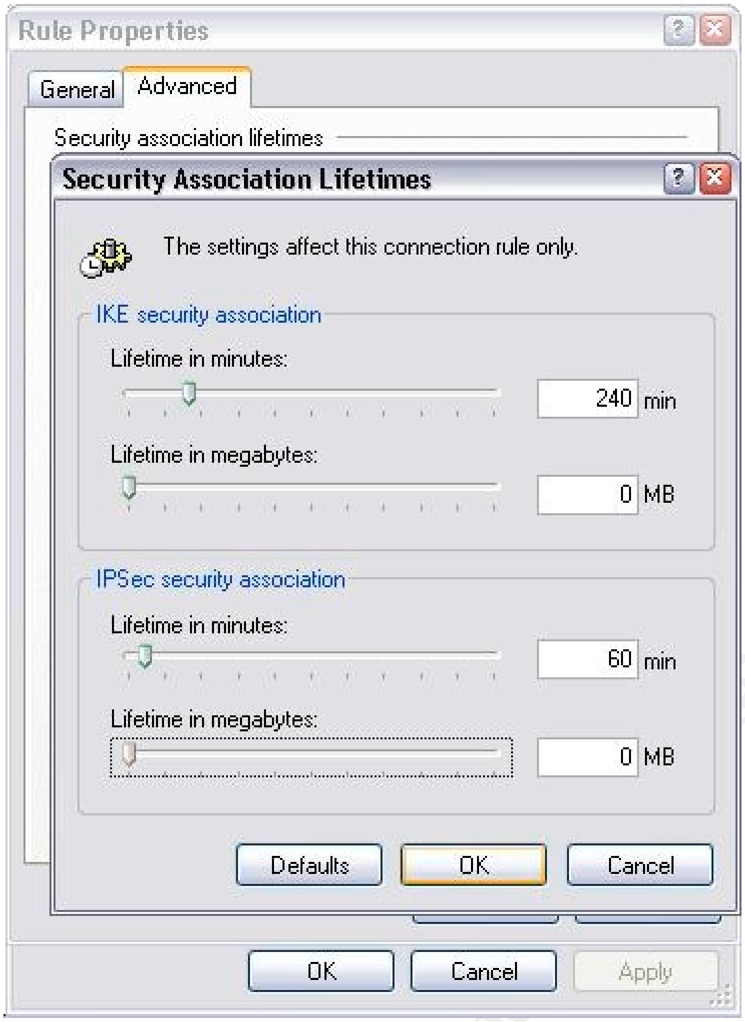
○ UDP encapsulation to port: 2746

☐ Open on start-up

☐ Deny split tunneling

Figure 9 - Advanced options

Click on ``Advanced" tab at the top of the box. Make sure ``Deny split tunneling" is unselected".


Rule Properties
General
Advanced
Security association lifetimes
Security Association Lifetimes
Defaults
OK
Cancel
OK
Cancel
Apply

The settings affect this connection rule only.

IKE security association

Lifetime in minutes: 240 min

Lifetime in megabytes: 0 MB

IPSec security association

Lifetime in minutes: 60 min

Lifetime in megabytes: 0 MB

Figure 10 - Advanced options (continued)

Click on ``Settings''. Make sure ``IPsec security association - Lifetime in megabytes'' set to zero. Click OK, and OK again.

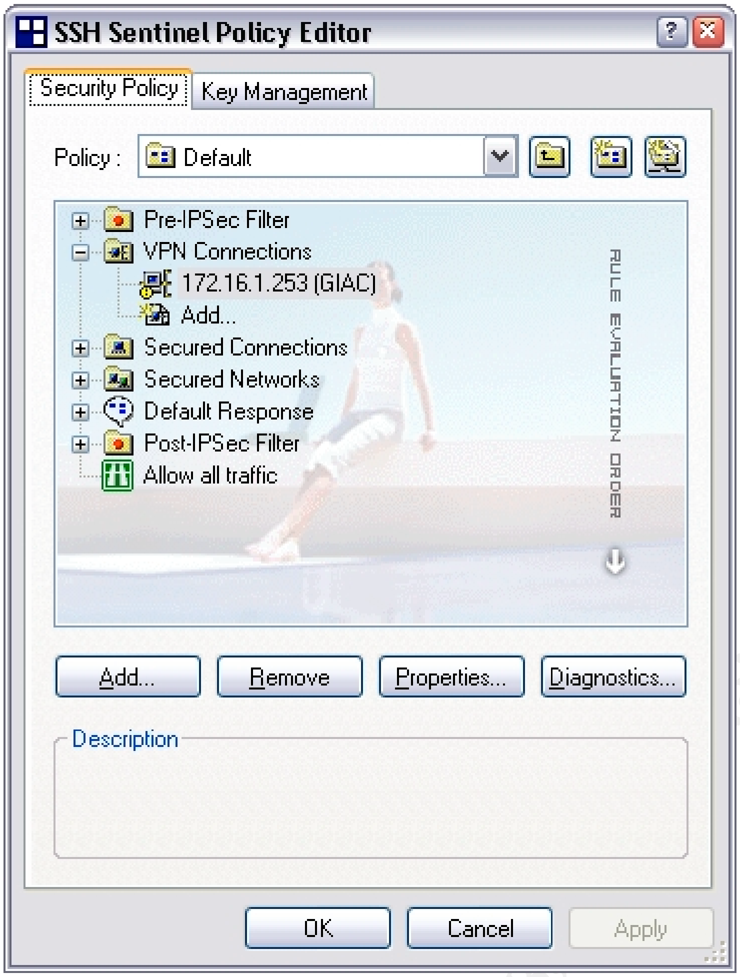


Figure 11 - VPN connection configured

Just click on ``Apply". The ``Diagnostics" will try to test if is possible to make an IPsec connection. With the FreeS/WAN and SSH Sentinel configuration used in this paper, the diagnostics will pass. To establish the real IPsec connection, right-click on SSH Sentinel icon in the taskbar then select ``Select VPN (172.16.1.253 (GIAC)" option.

Assignment 3 - Verify the Firewall Policy

To verify if the firewall rules is in accordance with the security policy defined for GIAC Enterprises, it is necessary to conduct a complete audit on primary and internal firewalls. This audit includes rules verification, vulnerability and penetration tests. Should be always conducted if the security policy changes or a rule is being added to the system.

3.1 - Audit Plan

The audit is organized in two phases, the primary firewall is audited in phase one, to verify if the system is capable to protect the DMZ and Web Cluster segments againts different scan types. Also, the system must be capable to block all traffic not allowed in or out DMZ and Web Cluster segments. The Same process is done in phase two, auditing the internal firewall, which must be capable to protect the internal network from both intenal and external attacks.

3.1.1 - Audit Cost

The audit cost includes labor and software cost. Labor costs are limited on employees necessary to plan and execute the audit; one system administrator and one security analyst is enough. On software costs, only one tool used is proprietary, the Eeye's Retina vulnerability scanner http://www.eeye.com, and its cost is aroud US $2600,00 for a 64 IP pack. All other tools used are free software based tools.

3.1.2 - Tools used

Nmap - http://www.insecure.org - Is a very good security auditing tool. It is used in the audit process to conduct a ICMP, UDP and TCP scan on GIAC's network to verify firewall rules. Nmap is a free software, the source code is available under the terms of the GNU GPL.

hping - http://www.hping.org - Is a command line packet assembler and analyzer, and it is used for firewall testing due its ability on packet crafting, specially fragmented packets.

tcpdump - http://www.tcpdump.org - Tcpdump is a powerful tool that allows packet sniffing and make statistical analysis on the output produced. It is used in the audit process to verify if the firewall rules are correct and ensure there's no packet leaking in our out GIAC's network.

Eeye's Retina Network Scanner - http://www.eeye.com - Retina can scan every machine on a network, including a variety of operating systems (Windows, Unix, GNU/Linux), networked devices (firewalls, routers, etc.), databases and third-party applications. Retina produces a very useful and comprehensive report that details all vulnerabilities, corrective actions and fixes.

3.2 - Phase 1 - primary firewall audit

3.2.1 - External Scan

This test is conducted using a notebook running Red Hat GNU/Linux posing as an attacker placed on the perimeter segment, using the IP address 172.16.1.252. Tcpdump is used on every machine on DMZ and Web Cluster segments. Entire output logs cannot be reproduced in this paper, only the most important excerpts is shown.

Vulnerability scan/Penetration test - Eeye's Retina vulnerability scan didn't found any vulnerability related to the firewall. No open ports were found. It is not scope of this test to check vulnerabilities on DMZ, Web Cluster, Services-Management, Internal Users/RoadWarriors, Database-Applications and IDS segments. Vulnerability scan on these segments is done in another audit phase, but it's not discussed in this document.

3.2.1.1 - SYN scan on DMZ and Web Cluster segments

This scan is used to verify if only the services defined on section 1.8 are visible from a external network. Nmap is used to do this task. Tcpdump and NETFILTER/IPTABLES firewall logs are used just to ensure everything happened as expected.

```
# nmap -v -g 25 -sS -sR -p 1-65535 172.16.1.0/24

Starting nmap 3.45 ( http://www.insecure.org/nmap/ ) at 2003-
09-24 23:29 BRT

Host 172.16.1.1 appears to be down, skipping it.
[snip]

Host 172.16.1.5 appears to be up ... good.
Initiating SYN Stealth Scan against 172.16.1.5 at 23:30
Adding open port 25/tcp
The SYN Stealth Scan took 45 seconds to scan 65535 ports.
Initiating RPCGrind Scan against 172.16.1.5 at 23:31
The RPCGrind Scan took 2 seconds to scan 1 ports.
Interesting ports on 172.16.1.5:
(The 65534 ports scanned but not shown below are in state:
filtered)
PORT    STATE  SERVICE VERSION
25/tcp  open smtp

Host 172.16.1.6 appears to be up ... good.
Initiating SYN Stealth Scan against 172.16.1.6 at 23:31
Adding open port 25/tcp
The SYN Stealth Scan took 45 seconds to scan 65535 ports.
Initiating RPCGrind Scan against 172.16.1.6 at 23:32
The RPCGrind Scan took 2 seconds to scan 1 ports.
Interesting ports on 172.16.1.6:
(The 65534 ports scanned but not shown below are in state:
filtered)
PORT    STATE  SERVICE VERSION
25/tcp  open smtp

Host 172.16.1.7 appears to be down, skipping it.
[snip]

Host 172.16.1.33 appears to be up ... good.
Initiating SYN Stealth Scan against 172.16.1.33 at 23:35
Adding open port 80/tcp
Adding open port 443/tcp
The SYN Stealth Scan took 45 seconds to scan 65535 ports.
Initiating RPCGrind Scan against 172.16.1.33 at 23:36
The RPCGrind Scan took 4 seconds to scan 2 ports.
Interesting ports on 172.16.1.33:
(The 65533 ports scanned but not shown below are in state:
filtered)
PORT    STATE  SERVICE VERSION
```

```
80/tcp  open http
443/tcp  open https

Host 172.16.1.34 appears to be up ... good.
Initiating SYN Stealth Scan against 172.16.1.34 at 23:38
Adding open port 80/tcp
Adding open port 443/tcp
The SYN Stealth Scan took 45 seconds to scan 65535 ports.
Initiating RPCGrind Scan against 172.16.1.34 at 23:39
The RPCGrind Scan took 4 seconds to scan 2 ports.
Interesting ports on 172.16.1.34:
(The 65533 ports scanned but not shown below are in state:
filtered)
PORT    STATE  SERVICE VERSION
80/tcp  open http
443/tcp  open https

Host 172.16.1.35 appears to be up ... good.
Initiating SYN Stealth Scan against 172.16.1.35 at 23:39
Adding open port 80/tcp
Adding open port 443/tcp
The SYN Stealth Scan took 45 seconds to scan 65535 ports.
Initiating RPCGrind Scan against 172.16.1.35 at 23:40
The RPCGrind Scan took 4 seconds to scan 2 ports.
Interesting ports on 172.16.1.35:
(The 65533 ports scanned but not shown below are in state:
filtered)
PORT    STATE  SERVICE VERSION
80/tcp  open http
443/tcp  open https

Host 172.16.1.36 appears to be up ... good.
Initiating SYN Stealth Scan against 172.16.1.36 at 23:40
Adding open port 80/tcp
Adding open port 443/tcp
The SYN Stealth Scan took 45 seconds to scan 65535 ports.
Initiating RPCGrind Scan against 172.16.1.36 at 23:36
The RPCGrind Scan took 4 seconds to scan 2 ports.
Interesting ports on 172.16.1.36:
(The 65533 ports scanned but not shown below are in state:
filtered)
PORT    STATE  SERVICE VERSION
80/tcp  open http
443/tcp  open https

Host 172.16.1.37 appears to be down, skipping it.
```

[snip]

NETFILTER/IPTABLES log excerpt - The log shown below is just a excerpt from a log at laranja, the primary firewall. It demonstrates that a prohibited TCP packet was successfuly blocked/dropped.

```
Sep 24 23:39:05 laranja kernel: TCP packet dropped IN=eth0
OUT=eth1 SRC 172.16.1.252 DST=172.16.1.5 LEN=40 TOS=0x00
PREC=0x00 TTL=58 ID=39184 PROTO=TCP SPT=25 DPT=139 WINDOW=2048
RES=0X00 SYN URGP=0
```

The tcpdump running on every machine at DMZ and Webcluster detected only packets related to the services provided and allowed to external users.

3.2.1.2 - UDP scan on DMZ and Web Cluster segments

This scan is used to verify if only the services defined on section 1.8 are visible from a external network. Nmap is used to do this task. Tcpdump and NETFILTER/IPTABLES firewall logs are used just to ensure everything happened as expected.

```
# nmap -v -g 53 -sU -p 1-65535 172.16.1.0/24

Starting nmap 3.45 ( http://www.insecure.org/nmap/ ) at 2003-
09-25 00:27 BRT

Host 172.16.1.1 appears to be up ... good.
Initiating UDP Scan against 172.16.1.1 at 00:28
Adding open port 53/udp
The UDP Scan took 55 seconds to scan 65535 ports.
Interesting ports on 172.16.1.1:
PORT   STATE  SERVICE
53/udp open   domain

Host 172.16.1.2 appears to be down, skipping it.

Host 172.16.1.3 appears to be up ... good.
Initiating UDP Scan against 172.16.1.3 at 00:30
Adding open port 53/udp
The UDP Scan took 50 seconds to scan 65535 ports.
Interesting ports on 172.16.1.3:
PORT   STATE  SERVICE
53/udp open   domain

Host 172.16.1.4 appears to be down, skipping it.
```

[snip]

NETFILTER/IPTABLES log excerpt- The log shown below is just a excerpt from a log at laranja, the primary firewall. It demonstrates that a prohibited UDP packet was successfuly blocked/dropped.

```
Sep 25 00:28:32 laranja kernel: UDP packet dropped IN=eth0
OUT=eth1 SRC 172.16.1.252 DST=172.16.1.3 LEN=28 TOS=0x00
PREC=0x00 TTL=36 ID=9184 PROTO=UDP SPT=53 DPT=135 LEN=8
```

The tcpdump running on every machine at DMZ and Web Cluster detected only packets related to the services provided and allowed to external users.

3.2.1.3 - FIN scan on DMZ and Web Cluster segments

This scan is used to detect if the firewall is vulnerable to a FIN scan. The idea behind a FIN scan, is not based on a NEW connection, but in sending RST/FIN to a server to probe ports. Closed ports reply to CLEAR CONNECT FIN bit set packet with RST bit set, while open port must ignore the packet.

```
# nmap -v -R -g 25 -sF -p 1-65535 172.16.1.0/24

Starting nmap 3.45 ( http://www.insecure.org/nmap/ ) at 2003-
09-25 00:47 BRT

Host 172.16.1.1 appears to be up ... good.
Initiating FIN Scan against 172.16.1.1 at 00:48

All 65535 scanned ports on 172.16.1.1 are: filtered

Host 172.16.1.2 appears to be down, skipping it.

Host 172.16.1.3 appears to be up ... good.
Initiating FIN Scan against 172.16.1.3 at 00:49

All 65535 scanned ports on 172.16.1.3 are: filtered

Host 172.16.1.4 appears to be down, skipping it.

Host 172.16.1.5 appears to be up ... good.
Initiating FIN Scan against 172.16.1.5 at 00:51

All 65535 scanned ports on 172.16.1.5 are: filtered

Host 172.16.1.6 appears to be up ... good.
Initiating FIN Scan against 172.16.1.6 at 00:53

All 65535 scanned ports on 172.16.1.3 are: filtered

Host 172.16.1.7 appears to be down, skipping it.
[snip]

host 172.16.1.33 appears to be up ... good.
Initiating FIN Scan against 172.16.1.33 at 01:05

All 65535 scanned ports on 172.16.1.33 are: filtered

host 172.16.1.34 appears to be up ... good.
Initiating FIN Scan against 172.16.1.34 at 01:07

All 65535 scanned ports on 172.16.1.34 are: filtered

host 172.16.1.35 appears to be up ... good.
Initiating FIN Scan against 172.16.1.35 at 01:09

All 65535 scanned ports on 172.16.1.35 are: filtered
```

```
host 172.16.1.36 appears to be up ... good.
Initiating FIN Scan against 172.16.1.36 at 01:11

All 65535 scanned ports on 172.16.1.36 are: filtered

Host 172.16.1.37 appears to be down, skipping it.
```

[snip]

NETFILTER/IPTABLES log excerpt - The log shown below is just a excerpt from a log at laranja, the primary firewall. It demonstrates that a prohibited TCP packet was successfuly blocked/dropped.

```
Sep 25 01:09:15 laranja kernel: TCP packet dropped IN=eth0
OUT=eth1 SRC 172.16.1.252 DST=172.16.1.1 LEN=40 TOS=0x00
PREC=0x00 TTL=50 ID=39184 PROTO=TCP SPT=25 DPT=493 WINDOW=3072
RES=0x00 FIN UGRP=0
```

The tcpdump running on every machine at DMZ and Web Cluster did not detected packets from nmap. The firewall rules worked as expected.

3.2.1.4 - Null scan on DMZ and Web Cluster segments

The null scan idea is to send crafted TCP packets with no flags set. The different error response depends upon the platform the receiving end is running.

```
# nmap -v -g 25 -sN -p 1-65535 172.16.1.0/24

Starting nmap 3.45 ( http://www.insecure.org/nmap/ ) at 2003-
09-25 01:30 BRT

Host 172.16.1.1 appears to be up ... good.
Initiating NULL Scan against 172.16.1.1 at 01:31

All 65535 scanned ports on 172.16.1.1 are: filtered

Host 172.16.1.2 appears to be down, skipping it.

Host 172.16.1.3 appears to be up ... good.
Initiating NULL Scan against 172.16.1.3 at 01:33

All 65535 scanned ports on 172.16.1.3 are: filtered

Host 172.16.1.4 appears to be down, skipping it.

Host 172.16.1.5 appears to be up ... good.
Initiating NULL Scan against 172.16.1.5 at 01:35

All 65535 scanned ports on 172.16.1.5 are: filtered

Host 172.16.1.6 appears to be up ... good.
Initiating NULL Scan against 172.16.1.6 at 01:37

All 65535 scanned ports on 172.16.1.3 are: filtered
```

```
Host 172.16.1.7 appears to be down, skipping it.
[snip]

host 172.16.1.33 appears to be up ... good.
Initiating NULL Scan against 172.16.1.33 at 01:39

All 65535 scanned ports on 172.16.1.33 are: filtered

host 172.16.1.34 appears to be up ... good.
Initiating NULL Scan against 172.16.1.34 at 01:40

All 65535 scanned ports on 172.16.1.34 are: filtered

host 172.16.1.35 appears to be up ... good.
Initiating NULL Scan against 172.16.1.35 at 01:42

All 65535 scanned ports on 172.16.1.35 are: filtered

host 172.16.1.36 appears to be up ... good.
Initiating NULL Scan against 172.16.1.36 at 01:44

All 65535 scanned ports on 172.16.1.36 are: filtered

Host 172.16.1.37 appears to be down, skipping it.
[snip]
```

NETFILTER/IPTABLES log excerpt - The log shown below is just a excerpt from a log at laranja, the primary firewall. It demonstrates that a prohibited TCP packet was successfuly blocked/dropped.

```
Sep 25 01:31:18 laranja kernel: TCP packet dropped IN=eth0
OUT=eth1 SRC 172.16.1.252 DST=172.16.1.36 LEN=40 TOS=0x00
PREC=0x00 TTL=55 ID=33184 PROTO=TCP SPT=25 DPT=561 WINDOW=4096
RES=0x00 UGRP=0
```

The tcpdump running on every machine at DMZ and Web Cluster did not detected packets from nmap. The firewall rules worked as expected.

3.2.1.5 - XMAS scan on DMZ and Web Cluster segments

The idea on Xmas scan is to send crafted TCP packets with flags FIN URGENT PUSH set. The different error response depends upon the plataform the receiving end is running.

```
# nmap -v -R -g 25 -sX -p 1-65535 172.16.1.0/24

Starting nmap 3.45 ( http://www.insecure.org/nmap/ ) at 2003-
09-25 02:10 BRT

Host 172.16.1.1 appears to be up ... good.
Initiating XMAS Scan against 172.16.1.1 at 02:11

All 65535 scanned ports on 172.16.1.1 are: filtered
```

```
Host 172.16.1.2 appears to be down, skipping it.

Host 172.16.1.3 appears to be up ... good.
Initiating XMAS Scan against 172.16.1.3 at 02:13

All 65535 scanned ports on 172.16.1.3 are: filtered

Host 172.16.1.4 appears to be down, skipping it.

Host 172.16.1.5 appears to be up ... good.
Initiating XMAS Scan against 172.16.1.5 at 02:15

All 65535 scanned ports on 172.16.1.5 are: filtered

Host 172.16.1.6 appears to be up ... good.
Initiating XMAS Scan against 172.16.1.6 at 02:18

All 65535 scanned ports on 172.16.1.3 are: filtered

Host 172.16.1.7 appears to be down, skipping it.
[snip]

host 172.16.1.33 appears to be up ... good.
Initiating XMAS Scan against 172.16.1.33 at 02:21

All 65535 scanned ports on 172.16.1.33 are: filtered

host 172.16.1.34 appears to be up ... good.
Initiating XMAS Scan against 172.16.1.34 at 02:24

All 65535 scanned ports on 172.16.1.34 are: filtered

host 172.16.1.35 appears to be up ... good.
Initiating XMAS Scan against 172.16.1.35 at 02:26

All 65535 scanned ports on 172.16.1.35 are: filtered

host 172.16.1.36 appears to be up ... good.
Initiating XMAS Scan against 172.16.1.36 at 02:29

All 65535 scanned ports on 172.16.1.36 are: filtered

Host 172.16.1.37 appears to be down, skipping it.
[snip]
```

NETFILTER/IPTABLES log excerpt - The log shown below is just an excerpt from a log at laranja, the primary firewall. It demonstrates that a prohibited TCP packet was successfuly blocked/dropped.

```
Sep 25 02:24:18 laranja kernel: TCP packet dropped IN=eth0
OUT=eth1 SRC 172.16.1.252 DST=172.16.1.34 LEN=40 TOS=0x00
```

```
PREC=0x00 TTL=52 ID=39184 PROTO=TCP SPT=25 DPT=713 WINDOW=4096
RES=0x00 URG PSH FIN UGRP=0
```

The tcpdump running on every machine at DMZ and Web Cluster did not detected packets from nmap. The firewall rules worked as expected.

3.2.1.6 - HPING fragment scan on DMZ and Web Cluster segments

The idea of this test is to verify if the primary firewall could be bypassed by sending fragmented packets. All machines at DMZ and Web Cluster segments have ssh enabled accepting connections from Services-Management subnet. The Primary firewall has a rule allowing only this specific subnet send packets to the TCP port 22, packets comming from other sources should be dropped. This test is only conducted to a single machine.

```
# hping2 -V --frag --data 40 --count 3 --syn -p 22 172.16.1.1

using eth0, addr: 172.16.1.252, MTU: 1500
HPING 172.16.1.1 (eth0 172.16.1.1): S set, 40 headers + 40
data bytes

--- 172.16.1.1 hping statistic ---
3 packets tramitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
```

NETFILTER/IPTABLES log excerpt - The log shown below is just an excerpt from a log at laranja, the primary firewall. It demonstrates that a prohibited TCP packet was successfuly blocked/dropped.

```
Sep 25 02:45:50 laranja kernel: TCP packet dropped IN=eth0
OUT=eth1 SRC 172.16.1.252 DST=172.16.1.1 LEN=40 TOS=0x00
PREC=0x00 TTL=63 ID=22 PROTO=TCP SPT=1442 DPT=22 WINDOW=512
RES=0x00 SYN UGRP=0
```

Tcpdump output at laranja, show the fragmented packet produced by HPING. That does not means the packet sucessfully traversed the firewall.

```
02:45:50.682803 172.16.1.252.1154 > 172.16.1.1.22: S [bad hdr
length] (frag 243:16@0+)
02:45:50.682810 172.16.1.252 > 172.16.1.1: tcp (frag
243:16@16+)
02:45:50.682811 172.16.1.252 > 172.16.1.1: tcp (frag
243:16@32+)
02:45:50.682813 172.16.1.252 > 172.16.1.1: tcp (frag
243:12@48)
02:45:51.669322 172.16.1.252.1155 > 172.16.1.1.22: S [bad hdr
length] (frag 243:16@0+)
02:45:51.669328 172.16.1.252 > 172.16.1.1: tcp (frag
243:16@16+)
02:45:51.669330 172.16.1.252 > 172.16.1.1: tcp (frag
243:16@32+)
02:45:51.669331 172.16.1.252 > 172.16.1.1: tcp (frag
243:12@48)
02:45:52.669398 172.16.1.252.1156 > 172.16.1.1.22: S [bad hdr
length] (frag 243:16@0+)
```

```
02:45:52.669403 172.16.1.252 > 172.16.1.1: tcp (frag
243:16@16+)
02:45:52.669405 172.16.1.252 > 172.16.1.1: tcp (frag
243:16@32+)
02:45:52.669407 172.16.1.252 > 172.16.1.1: tcp (frag
243:12@48)
```

3.3 - Phase 2 - internal firewall audit

In this phase, the audit process has a different perspective. Nmap result was used in phase one to verify if only the authorized services was available. In this phase, we assume nmap will produce no result at all, because it will spoof the IP source address of different machines at DMZ, Web Cluster, Management-Services, Internal Users/RoadWarriors, Database-Applications and IDS segments. Tcpdump and IPFILTER logs show the necessary information to verify the internal firewall.

3.3.1 - UDP and SYN scan to Management-Services segment

- Spoofing DMZ segment - The ``attacker'' is placed on Intermediary segment using the 172.16.1.242 IP address.

```
# nmap -v -g 1024 -e eth0 -S 172.16.1.1 -sU -p 1-65535
192.168.1.0/24
```

[snip]

```
# nmap -v -g 123 -e eth0 -S 172.16.1.1 -sU -p 1-65535
192.168.1.0/24
```

[snip]

```
# nmap -v -g 25 -e eth0 -S 172.16.1.5 -sS -sR -p 1-65535
192.168.1.0/24
```

[snip]

```
# nmap -v -g 1024 -e eth0 -S 172.16.1.5 -sS -sR -p 1-65535
192.168.1.0/24
```

[snip]

IPFILTER log excerpt - The log shown below is just an excerpt from a log at limao, the internal firewall. It demonstrates that a prohibited UDP and TCP packet was successfuly blocked/dropped.

```
25/09/2003 03:03:45.627326 fxp1 @23:40 b 172.16.1.1,1024 ->
192.168.1.1,54563 PR udp len 20 28 OUT
```

[snip]

```
25/09/2003 03:08:23.567358 fxp1 @23:40 b 172.16.1.1,123 ->
192.168.1.1,453 PR udp len 20 28 OUT
```

[snip]

```
25/09/2003 03:14:12.325321 fxp1 @23:40 b 172.16.1.5,25 ->
192.168.1.1,36647 PR tcp len 20 40 -S OUT
```

[snip]

Tcpdump running on all machines at Management-Services segment detected only packets allowed in by limao, such as NTP, SYSLOG, SMTP and Backup. The firewal rules worked as expected.

Tcpdump UDP log:

```
03:08:23.709762 172.16.1.1.123 > 192.168.1.1.123:  [len=0]
[|ntp]
03:08:23.817952 172.16.1.1.123 > 192.168.1.1.123:  [len=0]
[|ntp]

03:12:15.284675 172.16.1.1.1024 > 192.168.1.3.514: udp 0
03:12:48.282635 172.16.1.1.1024 > 192.168.1.3.514: udp 0
```

Tcpdump TCP log:

```
03:14:57.430614 172.16.1.5.25 > 192.168.1.4.25: S
2942190585:2942190585(0) win 4096
03:14:57.430801 192.168.1.4.25 > 172.16.1.5.25: S
2565819918:2565819918(0) ack 2942190586 win 16384
03:14:57.446856 172.16.1.5.25 > 192.168.1.4.25: R
2942190586:2942190586(0) win 0 (DF)

03:15:14.550351 172.16.5.1024 > 192.168.1.9.6101: S
2290694168:2290694168(0) win 4096
03:15:14.550418 192.168.1.9.6101 > 172.16.5.1024: S
3233918358:3233918358(0) ack 2290694169 win 5840  (DF)
03:15:14.550737 172.16.5.1024 > 192.168.1.9.6101: R
2290694169:2290694169(0) win 0 (DF)
```

- Spoofing Web Cluster segment - The ``attacker'' still on Intermediary segment, but now uses a different source address.

```
# nmap -v -g 1024 -e eth0 -S 172.16.1.33 -sU -p 1-65535
192.168.1.0/24
```

[snip]

```
# nmap -v -g 123 -e eth0 -S 172.16.1.33 -sU -p 1-65535
192.168.1.0/24
```

[snip]

```
# nmap -v -g 25 -e eth0 -S 172.16.1.33 -sS -sR -p 1-65535
192.168.1.0/24
```

[snip]

```
# nmap -v -g 1024 -e eth0 -S 172.16.1.33 -sS -sR -p 1-65535
192.168.1.0/24
```

[snip]

IPFILTER log excerpt - The log shown below is just an excerpt from a log at limao, the internal firewall. It demonstrates that a prohibited UDP and TCP packet was successfuly blocked/dropped.

```
25/09/2003 03:28:45.125336 fxp1 @23:40 b 172.16.1.33,1024 ->
192.168.1.1,14563 PR udp len 20 28 OUT
[snip]

25/09/2003 03:33:26.767353 fxp1 @23:40 b 172.16.1.33,123 ->
192.168.1.1,453 PR udp len 20 28 OUT
[snip]

25/09/2003 03:39:12.325321 fxp1 @23:40 b 172.16.1.33,25 ->
192.168.1.1,139 PR tcp len 20 40 -S OUT
[snip]

25/09/2003 03:39:12.325321 fxp1 @23:40 b 172.16.1.33,1024 ->
192.168.1.1,7 PR tcp len 20 40 -S OUT
[snip]
```

The tcpdump running at 172.16.1.1 detected packets allowed in by limao, such as packets to the NTP, SYSLOG and Backup servers. The firewall rules worked as expected.

Tcpdump UDP log:

```
03:34:23.709762 172.16.1.33.123 > 192.168.1.1.123:  [len=0]
[|ntp]
03:34:23.817952 172.16.1.33.123 > 192.168.1.1.123:  [len=0]
[|ntp]

03:28:55.284675 172.16.1.33.1024 > 192.168.1.3.514: udp 0
03:28:58.382635 172.16.1.33.1024 > 192.168.1.3.514: udp 0
```

Tcpdump TCP log:

```
03:41:14.550351 172.16.33.1024 > 192.168.1.9.6101: S
2690694161:2690694161(0) win 4096
03:41:14.550418 192.168.1.9.6101 > 172.16.33.1024: S
2234958358:2234958358(0) ack 2690694162 win 5840  (DF)
03:41:14.550737 172.16.33.1024 > 192.168.1.9.6101: R
2690694162:2690694162(0) win 0 (DF)
```

- Scanning from Internal Users/RoadWarriors segment - Now, the scan is done from the Internal Users/RoadWarriors segment. It's not necessary to spoof the source IP address in this test.

```
# nmap -v -sU -p 1-65535 192.168.1.0/24
Starting nmap 3.45 ( http://www.insecure.org/nmap/ ) at 2003-
09-25 04:18 BRT

Host 192.168.1.1 appears to be up ... good.
Initiating UDP Scan against 192.168.1.1 at 04:18
The UDP Scan took 55 seconds to scan 65535 ports.
Adding open port 123/udp
Interesting ports on 192.168.1.1:
```

```
(The 65534 ports scanned but not shown below are in state:
closed)
Port       State       Service
123/udp     open        ntp

Host 192.168.1.2 appears to be up ... good.
Initiating UDP Scan against 192.168.1.2 at 04:19
The UDP Scan took 55 seconds to scan 65535 ports.
Adding open port 137/udp
Adding open port 138/udp
Interesting ports on 192.168.1.2:
(The 65533 ports scanned but not shown below are in state:
closed)
Port       State       Service
137/udp     open        netbios-ns
138/udp     open        netbios-dgm

Host 192.168.1.3 appears to be up ... good.
Initiating UDP Scan against 192.168.1.3 at 04:20
The UDP Scan took 55 seconds to scan 65535 ports.
Adding open port 514/udp
Interesting ports on 192.168.1.4:
(The 65534 ports scanned but not shown below are in state:
closed)
Port       State       Service
514/udp     open        syslog

Host 192.168.1.4 appears to be up ... good.
Initiating UDP Scan against 192.168.1.4 at 04:22
The UDP Scan took 58 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.4 are: closed

Host 192.168.1.5 appears to be up ... good.
Initiating UDP Scan against 192.168.1.5 at 04:24
The UDP Scan took 55 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.5 are: closed

Host 192.168.1.6 appears to be up ... good.
Initiating UDP Scan against 192.168.1.6 at 04:25
The UDP Scan took 55 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.6 are: closed

Host 192.168.1.7 appears to be up ... good.
Initiating UDP Scan against 192.168.1.7 at 04:27
The UDP Scan took 55 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.7 are: closed

Host 192.168.1.8 appears to be up ... good.
Initiating UDP Scan against 192.168.1.8 at 04:28
The UDP Scan took 55 seconds to scan 65535 ports.
Adding open port 53/udp
```

```
Interesting ports on 192.168.1.8:
(The 65534 ports scanned but not shown below are in state:
closed)
Port       State      Service
53/udp     open       domain

Host 192.168.1.9 appears to be up ... good.
Initiating UDP Scan against 192.168.1.9 at 04:30
The UDP Scan took 55 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.9 are: closed

Host 192.168.1.10 appears to be up ... good.
Initiating UDP Scan against 192.168.1.10 at 04:31
The UDP Scan took 55 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.10 are: closed

Host 192.168.1.11 appears to be down, skipping it.
[snip]

# nmap -v -sS -sR -p 1-65535 192.168.1.0/24

Starting nmap 3.45 ( http://www.insecure.org/nmap/ ) at 2003-
09-25 04:35 BRT

Host 192.168.1.1 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.1 at 04:35
The SYN Stealth Scan took 36 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.1 are: closed

Host 192.168.1.2 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.2 at 04:37
Adding open port 139/tcp
Adding open port 389/tcp
Adding open port 515/tcp
The SYN Stealth Scan took 45 seconds to scan 65535 ports.
Initiating RPCGrind Scan against 192.168.1.2 at 04:39
The RPCGrind Scan took 6 seconds to scan 3 ports.
Interesting ports on 192.168.1.2:
(The 65532 ports scanned but not shown below are in state:
filtered)
PORT    STATE  SERVICE VERSION
139/tcp  open  netbios-ssn
389/tcp  open  ldap
515/tcp  open  printer

Host 192.168.1.3 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.3 at 04:40
The SYN Stealth Scan took 36 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.3 are: closed

Host 192.168.1.4 appears to be up ... good.
```

```
Initiating SYN Stealth Scan against 192.168.1.4 at 04:40
Adding open port 25/tcp
Adding open port 143/tcp
Adding open port 993/tcp
Adding open port 995/tcp
The SYN Stealth Scan took 45 seconds to scan 65535 ports.
Initiating RPCGrind Scan against 192.168.1.4 at 04:39
The RPCGrind Scan took 8 seconds to scan 4 ports.
Interesting ports on 192.168.1.4:
(The 65532 ports scanned but not shown below are in state:
filtered)
PORT    STATE  SERVICE VERSION
25/tcp     open        smtp
143/tcp    open        imap2
993/tcp    open        imaps
995/tcp    open        pop3s

Host 192.168.1.5 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.5 at 04:42
Adding open port 80/tcp
Adding open port 443/tcp
The SYN Stealth Scan took 47 seconds to scan 65535 ports.
Initiating RPCGrind Scan against 192.168.1.5 at 04:44
The RPCGrind Scan took 6 seconds to scan 3 ports.
Interesting ports on 192.168.1.5:
(The 65532 ports scanned but not shown below are in state:
filtered)
PORT    STATE  SERVICE VERSION
80/tcp    open        http
443/tcp   open        https

Host 192.168.1.6 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.6 at 04:44
Adding open port 3128/tcp
The SYN Stealth Scan took 45 seconds to scan 65535 ports.
Initiating RPCGrind Scan against 192.168.1.6 at 04:45
The RPCGrind Scan took 1 seconds to scan 1 ports.
Interesting ports on 192.168.1.6:
(The 65532 ports scanned but not shown below are in state:
filtered)
PORT    STATE  SERVICE VERSION
3128/tcp   open       squid-http

Host 192.168.1.7 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.7 at 04:46
The SYN Stealth Scan took 36 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.7 are: closed

Host 192.168.1.8 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.8 at 04:47
The SYN Stealth Scan took 36 seconds to scan 65535 ports.
```

```
All 65535 scanned ports on 192.168.1.8 are: closed

Host 192.168.1.9 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.9 at 04:48
Adding open port 6101/tcp
The SYN Stealth Scan took 45 seconds to scan 65535 ports.
Initiating RPCGrind Scan against 192.168.1.9 at 04:49
The RPCGrind Scan took 1 seconds to scan 1 ports.
Interesting ports on 192.168.1.9:
(The 65532 ports scanned but not shown below are in state:
filtered)
PORT    STATE  SERVICE VERSION
6101/tcp   open       VeritasBackupExec

Host 192.168.1.10 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.10 at 04:49
The SYN Stealth Scan took 36 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.10 are: closed

Host 192.168.1.11 appears to be down, skipping it.
```

[snip]

IPFILTER log excerpt - The log shown below is just an excerpt from a log at limao, the internal firewall. It demonstrates that a prohibited UDP and TCP packet was successfuly blocked/dropped.

```
25/09/2003 04:18:35.424336 fxp2 @16:50 b 192.168.2.33,1024 ->
192.168.1.1,14563 PR udp len 20 28 OUT
```

[snip]

```
25/09/2003 04:35:02.325321 fxp2 @16:50 b 192.168.2.33,1024 ->
192.168.1.1,7 PR tcp len 20 40 -S OUT
```

[snip]

Tcpdump was set up to run on every machine on Services-Management segment. The logs shown below are the result of all data collected by tcpdump on every machine. The firewall rules worked as expected.

Tcpdump UDP log:

```
04:18:13.509751 192.168.2.33.123 > 192.168.1.1.123:  [len=0]
[|ntp]
04:18:13.617941 192.168.2.33.123 > 192.168.1.1.123:  [len=0]
[|ntp]

04:19:55.284675 192.168.2.33.1024 > 192.168.1.2.137: udp 0
04:19:55.382635 192.168.2.33.1024 > 192.168.1.2.138: udp 0

04:20:07.253119 192.168.2.33.1024 > 192.168.1.3.514: udp 0

04:28:35.655119 192.168.2.33.1024 > 192.168.1.8.53: 0 [0q] (0)
```

Tcpdump TCP log:

```
04:37:14.882594 192.168.2.33.33491 > 192.168.1.2.139: S
2858195700:2858195700(0) win 4096
04:37:14.882648 192.168.1.2.139 > 192.168.2.33.33491: S
3989816218:3989816218(0) ack 2858195701 win 5840  (DF)
04:37:14.883124 192.168.2.33.33491 > 192.168.1.2.139: R
2858195701:2858195701(0) win 0 (DF)

04:37:53.432597 192.168.2.33.55174 > 192.168.1.2.389: S
3089825976:3089825976(0) win 4096
04:37:53.432661 192.168.1.2.389 > 192.168.2.33.55174: S
4147831046:4147831046(0) ack 3089825977 win 5840  (DF)
04:37:53.433166 192.168.2.33.55174 > 192.168.1.2.389: R
3089825977:3089825977(0) win 0 (DF)

04:38:09.650083 192.168.2.33.61308 > 192.168.1.2.515: S
762112328:762112328(0) win 2048
04:38:09.650149 192.168.1.2.515 > 192.168.2.33.61308: S
4202088634:4202088634(0) ack 762112329 win 5840  (DF)
04:38:09.704128 192.168.2.33.61308 > 192.168.1.2.515: R
762112329:762112329(0) win 0 (DF)

04:40:03.625985 192.168.2.33.51295 > 192.168.1.4.25: S
2287600834:2287600834(0) win 4096
04:40:03.626046 192.168.1.4.25 > 192.168.2.33.51295: S
1121128305:1121128305(0) ack 2287600835 win 5840  (DF)
04:40:03.626343 192.168.2.33.51295 > 192.168.1.4.25: R
2287600835:2287600835(0) win 0 (DF)

04:40:31.555438 192.168.2.33.37555 > 192.168.1.4.143: S
546974343:546974343(0) win 1024
04:40:31.555499 192.168.1.4.143 > 192.168.2.33.37555: S
1324566506:1324566506(0) ack 546974344 win 5840  (DF)
04:40:31.555811 192.168.2.33.37555 > 192.168.1.4.143: R
546974344:546974344(0) win 0 (DF)

04:40:54.052220 192.168.2.33.53423 > 192.168.1.4.993: S
1828771986:1828771986(0) win 3072
04:40:54.052285 192.168.1.4.993 > 192.168.2.33.53423: S
1400220198:1400220198(0) ack 1828771987 win 5840  (DF)
04:40:54.088475 192.168.2.33.53423 > 192.168.1.4.993: R
1828771987:1828771987(0) win 0 (DF)

04:41:19.475779 192.168.2.33.36696 > 192.168.1.4.995: S
4233342671:4233342671(0) win 2048
04:41:19.475834 192.168.1.4.995 > 192.168.2.33.36696: S
1413601442:1413601442(0) ack 4233342672 win 5840  (DF)
04:41:19.476358 192.168.2.33.36696 > 192.168.1.4.995: R
4233342672:4233342672(0) win 0 (DF)

04:42:09.545504 192.168.2.33.37401 > 192.168.1.5.80: S
3533186183:3533186183(0) win 3072
```

```
04:42:09.545573 192.168.1.5.80 > 192.168.2.33.37401: S
1646102485:1646102485(0) ack 3533186184 win 5840  (DF)
04:42:09.576787 192.168.2.33.37401 > 192.168.1.5.80: R
3533186184:3533186184(0) win 0 (DF)

04:42:33.095635 192.168.2.33.59898 > 192.168.1.5.443: S
2488501227:2488501227(0) win 4096
04:42:33.095697 192.168.1.5.443 > 192.168.2.33.59898: S
1676431224:1676431224(0) ack 2488501228 win 5840  (DF)
04:42:33.139242 192.168.2.33.59898 > 192.168.1.5.443: R
2488501228:2488501228(0) win 0 (DF)

04:44:28.635893 192.168.2.33.58843 > 192.168.1.6.3128: S
393878014:393878014(0) win 4096
04:44:28.635960 192.168.1.6.3128 > 192.168.2.33.58843: S
1785614711:1785614711(0) ack 393878015 win 5840  (DF)
04:44:28.681100 192.168.2.33.58843 > 192.168.1.6.3128: R
393878015:393878015(0) win 0 (DF)

04:48:30.677707 192.168.2.33.56649 > 192.168.1.9.6101: S
1370770706:1370770706(0) win 2048
04:48:30.677779 192.168.1.9.6101 > 192.168.2.33.56649: S
1915055418:1915055418(0) ack 1370770707 win 5840  (DF)
04:48:30.711910 192.168.2.33.56649 > 192.168.1.9.6101: R
1370770707:1370770707(0) win 0 (DF)
```

- Scanning from the Database-Applications segment - This test is very similar the test done above, and doesn't need to spoof the source IP address.

```
# nmap -v -sU -p 1-65535 192.168.1.0/24

Starting nmap 3.45 ( http://www.insecure.org/nmap/ ) at 2003-
09-25 05:15 BRT

Host 192.168.1.1 appears to be up ... good.
Initiating UDP Scan against 192.168.1.1 at 04:15
The UDP Scan took 55 seconds to scan 65535 ports.
Adding open port 123/udp
Interesting ports on 192.168.1.1:
(The 65534 ports scanned but not shown below are in state:
closed)
Port       State       Service
123/udp     open        ntp

Host 192.168.1.2 appears to be up ... good.
Initiating UDP Scan against 192.168.1.2 at 05:18
The UDP Scan took 58 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.2 are: closed

Host 192.168.1.3 appears to be up ... good.
Initiating UDP Scan against 192.168.1.3 at 05:20
```

```
The UDP Scan took 55 seconds to scan 65535 ports.
Adding open port 514/udp
Interesting ports on 192.168.1.3:
(The 65534 ports scanned but not shown below are in state:
closed)
Port       State       Service
514/udp     open        syslog

Host 192.168.1.4 appears to be up ... good.
Initiating UDP Scan against 192.168.1.4 at 05:22
The UDP Scan took 58 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.4 are: closed

Host 192.168.1.5 appears to be up ... good.
Initiating UDP Scan against 192.168.1.5 at 05:24
The UDP Scan took 55 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.5 are: closed

Host 192.168.1.6 appears to be up ... good.
Initiating UDP Scan against 192.168.1.6 at 05:25
The UDP Scan took 55 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.6 are: closed

Host 192.168.1.7 appears to be up ... good.
Initiating UDP Scan against 192.168.1.7 at 05:27
The UDP Scan took 55 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.7 are: closed

Host 192.168.1.8 appears to be up ... good.
Initiating UDP Scan against 192.168.1.8 at 05:28
The UDP Scan took 55 seconds to scan 65535 ports.
Adding open port 53/udp
Interesting ports on 192.168.1.8:
(The 65534 ports scanned but not shown below are in state:
closed)
Port       State       Service
53/udp     open        domain

Host 192.168.1.9 appears to be up ... good.
Initiating UDP Scan against 192.168.1.9 at 05:30
The UDP Scan took 55 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.9 are: closed

Host 192.168.1.10 appears to be up ... good.
Initiating UDP Scan against 192.168.1.10 at 05:31
The UDP Scan took 55 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.10 are: closed

Host 192.168.1.11 appears to be down, skipping it.
```

[snip]

```
# nmap -v -g 1024 -sS -sR -p 1-65535 192.168.1.0/24

Starting nmap 3.45 ( http://www.insecure.org/nmap/ ) at 2003-
09-25 05:40 BRT

Host 192.168.1.1 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.1 at 05:40
The SYN Stealth Scan took 36 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.1 are: closed

Host 192.168.1.2 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.2 at 05:42
The SYN Stealth Scan took 36 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.2 are: closed

Host 192.168.1.3 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.3 at 05:43
The SYN Stealth Scan took 36 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.3 are: closed

Host 192.168.1.4 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.4 at 05:45
The SYN Stealth Scan took 36 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.4 are: closed

Host 192.168.1.5 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.5 at 05:47
The SYN Stealth Scan took 36 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.5 are: closed

Host 192.168.1.6 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.6 at 05:49
The SYN Stealth Scan took 36 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.6 are: closed

Host 192.168.1.7 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.7 at 05:51
The SYN Stealth Scan took 36 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.7 are: closed

Host 192.168.1.8 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.8 at 05:52
The SYN Stealth Scan took 36 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.8 are: closed

Host 192.168.1.9 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.9 at 05:54
Adding open port 6101/tcp
The SYN Stealth Scan took 45 seconds to scan 65535 ports.
Initiating RPCGrind Scan against 192.168.1.9 at 05:55
The RPCGrind Scan took 1 seconds to scan 1 ports.
```

```
Interesting ports on 192.168.1.9:
(The 65532 ports scanned but not shown below are in state:
filtered)
PORT    STATE  SERVICE VERSION
6101/tcp   open       VeritasBackupExec

Host 192.168.1.10 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.10 at 05:56
The SYN Stealth Scan took 36 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.10 are: closed

Host 192.168.1.11 appears to be down, skipping it.
```

[snip]

IPFILTER log excerpt - The log shown below is just an excerpt from a log at limao, the internal firewall. It demonstrates that a prohibited UDP and TCP packet was successfuly blocked/dropped.

```
25/09/2003 05:15:45.424131 fxp3 @6:70 b 192.168.3.126,1024 ->
192.168.1.1,666 PR udp len 20 28 OUT
```

[snip]

```
25/09/2003 05:40:32.525529 fxp3 @6:70 b 192.168.3.126,1024 ->
192.168.1.1,139 PR tcp len 20 40 -S OUT
```

[snip]

Tcpdump was set up to run on every machine on Services-Management segment. The logs shown below are the result of all data collected by tcpdump on every machine. The firewall rules worked as expected.

Tcpdump UDP log:

```
05:15:13.109722 192.168.3.126.123 > 192.168.1.1.123:  [len=0]
[|ntp]

05:20:07.253119 192.168.3.126.1024 > 192.168.1.3.514: udp 0

05:28:35.655119 192.168.3.126.1024 > 192.168.1.8.53: 0 [0q]
(0)
```

Tcpdump TCP log - Tcpdump didn't detected any TCP packet from Services-Management segment.

- Scanning from the IDS segment - It is expected only DNS and NTP services to be open.

```
# nmap -v -sU -p 1-65535 192.168.1.0/24

Starting nmap 3.45 ( http://www.insecure.org/nmap/ ) at 2003-
09-25 06:20 BRT

Host 192.168.1.1 appears to be up ... good.
Initiating UDP Scan against 192.168.1.1 at 06:20
The UDP Scan took 55 seconds to scan 65535 ports.
Adding open port 123/udp
```

```
Interesting ports on 192.168.1.1:
(The 65534 ports scanned but not shown below are in state:
closed)
Port       State       Service
123/udp     open        ntp

Host 192.168.1.2 appears to be up ... good.
Initiating UDP Scan against 192.168.1.2 at 06:22
The UDP Scan took 58 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.2 are: closed

Host 192.168.1.3 appears to be up ... good.
Initiating UDP Scan against 192.168.1.3 at 06:24
The UDP Scan took 58 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.3 are: closed

Host 192.168.1.4 appears to be up ... good.
Initiating UDP Scan against 192.168.1.4 at 06:27
The UDP Scan took 58 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.4 are: closed

Host 192.168.1.5 appears to be up ... good.
Initiating UDP Scan against 192.168.1.5 at 06:28
The UDP Scan took 55 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.5 are: closed

Host 192.168.1.6 appears to be up ... good.
Initiating UDP Scan against 192.168.1.6 at 06:29
The UDP Scan took 55 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.6 are: closed

Host 192.168.1.7 appears to be up ... good.
Initiating UDP Scan against 192.168.1.7 at 06:30
The UDP Scan took 55 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.7 are: closed

Host 192.168.1.8 appears to be up ... good.
Initiating UDP Scan against 192.168.1.8 at 06:31
The UDP Scan took 55 seconds to scan 65535 ports.
Adding open port 53/udp
Interesting ports on 192.168.1.8:
(The 65534 ports scanned but not shown below are in state:
closed)
Port       State       Service
53/udp     open        domain

Host 192.168.1.9 appears to be up ... good.
Initiating UDP Scan against 192.168.1.9 at 06:32
The UDP Scan took 55 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.9 are: closed
```

```
Host 192.168.1.10 appears to be up ... good.
Initiating UDP Scan against 192.168.1.10 at 06:33
The UDP Scan took 55 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.10 are: closed

Host 192.168.1.11 appears to be down, skipping it.
[snip]

# nmap -v -g 1024 -sS -sR -p 1-65535 192.168.1.0/24

Starting nmap 3.45 ( http://www.insecure.org/nmap/ ) at 2003-
09-25 06:41 BRT

Host 192.168.1.1 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.1 at 06:41
The SYN Stealth Scan took 36 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.1 are: closed

Host 192.168.1.2 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.2 at 06:42
The SYN Stealth Scan took 36 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.2 are: closed

Host 192.168.1.3 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.3 at 06:42
The SYN Stealth Scan took 36 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.3 are: closed

Host 192.168.1.4 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.4 at 06:43
The SYN Stealth Scan took 36 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.4 are: closed

Host 192.168.1.5 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.5 at 06:44
The SYN Stealth Scan took 36 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.5 are: closed

Host 192.168.1.6 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.6 at 06:45
The SYN Stealth Scan took 36 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.6 are: closed

Host 192.168.1.7 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.7 at 06:46
The SYN Stealth Scan took 36 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.7 are: closed

Host 192.168.1.8 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.8 at 06:47
The SYN Stealth Scan took 36 seconds to scan 65535 ports.
```

```
All 65535 scanned ports on 192.168.1.8 are: closed

Host 192.168.1.9 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.9 at 06:48
The SYN Stealth Scan took 36 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.9 are: closed

Host 192.168.1.10 appears to be up ... good.
Initiating SYN Stealth Scan against 192.168.1.10 at 06:49
The SYN Stealth Scan took 36 seconds to scan 65535 ports.
All 65535 scanned ports on 192.168.1.10 are: closed

Host 192.168.1.11 appears to be down, skipping it.
```

[snip]

IPFILTER log excerpt - The log shown below is just an excerpt from a log at limao, the internal firewall. It demonstrates that a prohibited UDP and TCP packet was successfuly blocked/dropped.

```
25/09/2003 06:20:45.224632 fxp4 @5:90 b 192.168.255.64,1024 ->
192.168.1.1,53 PR udp len 20 28 OUT
```

[snip]

```
25/09/2003 06:41:03.132351 fxp4 @5:90 b 192.168.255.64,1024 ->
192.168.1.1,80 PR tcp len 20 40 -S OUT
```

[snip]

Tcpdump was set up to run on every machine on Services-Management segment. The logs shown below are the result of all data collected by tcpdump on every machine. The firewall rules worked as expected.

Tcpdump UDP log:

```
05:50:29.109722 192.168.255.64.123 > 192.168.1.1.123:  [len=0]
[|ntp]

06:03:35.655119 192.168.255.64.1024 > 192.168.1.8.53: 0 [0q]
(0)
```

Tcpdump TCP log - Tcpdump didn't detected any TCP packet from Services-Management segment.

3.3.2 - UDP and SYN scan to Internal Users/RoadWarriors segment

The Internal Users/RoadWarriors segment does not accept packets from non related or established connections originated inside this segment. That means all scan attempts done from DMZ, Web Cluster, Services-Management, Database-Applications and IDS segments failed. A tcpdump running on a machine placed at Internal Users/RoadWarriors segment didn't detected any packets. The firewall rules worked as expected.

3.3.3 - UDP and SYN scan to Database-Applications segment

This segment just offer services to the Web Cluster segment. All scan attempt from DMZ, Services-Management, Internal Users/RoadWarriors, Database-Applications and IDS segments failed. The firewall rules worked as expected.

- Scanning from the Web Cluster segment - In this test, it is necessary to spoof the source IP address of a machine at Web Cluster segment. Tcpdump and IPFILTER logs is used to verify if the firewall rules are correct.

```
# nmap -v -g 1024 -e eth0 -S 172.16.1.33 -sU -p 1-65535
192.168.3.0/24
```

[snip]

```
# nmap -v -g 1024 -e eth0 -S 172.16.1.33 -sS -sR -p 1-65535
192.168.3.0/24
```

[snip]

IPFILTER log excerpt - The log shown below is just an excerpt from a log at limao, the internal firewall. It demonstrates that a prohibited UDP and TCP packet was successfuly blocked/dropped.

```
25/09/2003 07:23:45.117327 fxp3 @16:80 b 172.16.1.33,1024 ->
192.168.3.3,135 PR udp len 20 28 OUT
```

[snip]

```
25/09/2003 07:56:42.324394 fxp3 @16:80 b 172.16.1.33,1024 ->
192.168.3.3,80 PR tcp len 20 40 -S OUT
```

[snip]

Tcpdump running on every machine at Database-Applications segment detected only packets allowed in by limao. The firewall rules worked as expected.

Tcpdump UDP log - Tcpdump didn't detected any UDP packet from Services-Management segment.

Tcpdump TCP log:

```
07:55:39.428627 172.16.1.33.1024 > 192.168.3.3.8443: S
14333684:14333684(0) win 3072
07:55:39.428699 192.168.3.3.8443 > 172.16.1.33.1024: S
1466646096:1466646096(0) ack 14333685 win 5840  (DF)
07:55:39.470716 172.16.1.33.1024 > 192.168.3.3.8443: R
14333685:14333685(0) win 0 (DF)
```

3.3.4 - UDP and SYN scan to IDS segment

The IDS segment has the same behavior as Internal Users/RoadWarriors segment. It does not offer any service or accept packets from non related or established connections originated inside IDS segment. Scan attempts from DMZ, Web Cluster, Services-Management, Database-Applications and Internal Users/RoadWarriors segment failed. The firewall rules worked as expected.

3.4 - Audit Report considerations

IPTABLES and IPFILTER does a very good job as a packet filter. Both are stateful firewalls, imune from the most common firewall problems, such as SYN, ACK, FIN, NULL and XMAS scans and fragmentation.

In all tests perfomed, NETFILTER/IPTABLES and IPFILTER worked without any flaw. This may be different if other software requirements were necessary, such as support for H.323 or other special multimedia protocol.

3.5 - Recomendations and improvements

High Availability would be a plus to the primary firewall, avoiding the single point of failure problem. To facilitate the System Administrator job, a GUI-based rules editor might be necessary as the rules file grows.

Assignment 4 - Design Under Fire

Brad Tauer's practical design will be used in this assignment. The original document could be found at http://www.giac.org/practical/GCFW/Brad_Tauer.pdf.

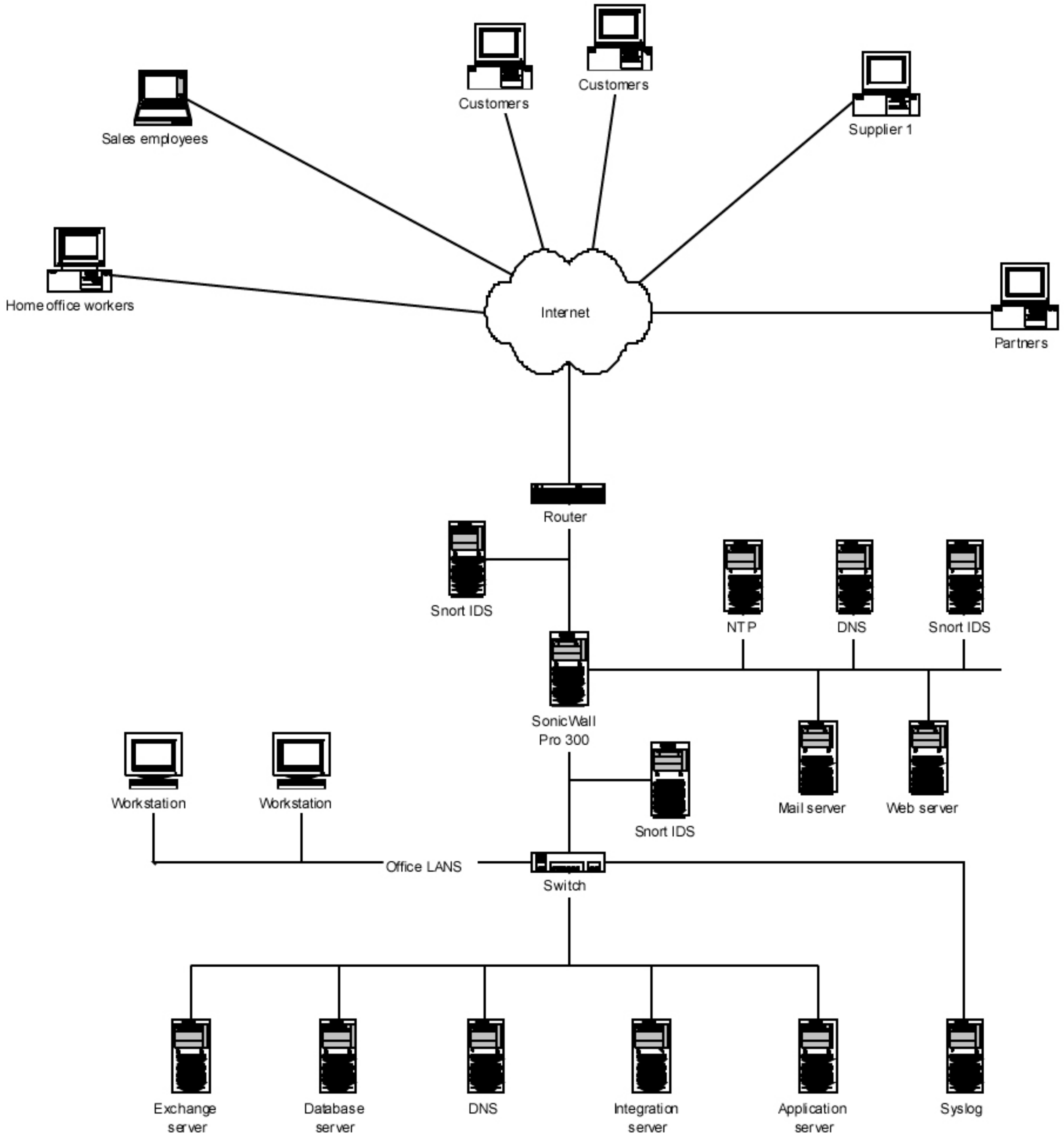


Figure 12- Brad Tauer's network design

4.1 - Attack Against the Firewall

Brad chose the SonicWall 200 firewall appliance because thinks its easier and better running a proprietary firewall than running a free software based firewall.

SonicWall Security Advisories - http://www.sonicwall.com/support/securityadvisories.html lists only 3 vulnerabilities since 02/2002. All vulnerabilities are considered too old if we consider the fact Brad's desgin and implementation period, March 2003. Its unlike that Brad implemented a firewall appliance with a 8 month-old vulnerability.

- Multiple Vulnerabilities In OpenSSL, 7/30/2002 -
- Content Blocking Script Injection Vulnerability Advisory, 5/23/2002
- CERT Advisory CA-2002-03 SNMP, 2/19/2002

But wait! Searching bugtraq archives I've found a Denial of Service vulnerability on Sonicwall PRO. This is discussed at http://www.securityfocus.com/archive1/319712.

The message describes a buffer overflow vulnerability when a very large HTTP POST is sent to the Sonicwall or machines under the firewall protection. The vulnerability was confirmed using 2 Nessus plugins: www_too_long_post and alibaba_overflow.

4.1.1 - Attacking the firewall

To attack the firewall, it is necessary run the Nessus network scanner with the appropriate plugin, as shown below:

```
# nasl -t a.b.c.d* www_too_long_post.nasl
```

or

```
# nasl -t a.b.c.d* alibaba_overflow.nasl
```

* where a.b.c.d is the internal Web server.

If Brad installed the appropriate patches before this attack was done, the attack will fail. Otherwise, its expected the firewall reboot.

4.2 - Distributed Denial of Service

4.2.1 - The attack

After compromising 50 GNU/Linux boxes, we lauch a DDoS SYN flood attackg against a webserver in XYZ design. The attack consists using a modified version of juno, a very powerful SYN flood DoS tool. The packets have a spoofed source IP address, dificulting the investigation process of finding the origin of the attack. It is expected total consumption of both firewall processing and bandwidth capacity. Each compromised box is capable to lauch a high-rate packet attack, such as 70,000 packets/s. Most stateful firewalls cannot maintain the stateful table useful for a long period.

A SYN flood attack consists in sending a huge amount of TCP packets with the SYN flag set and then nothing else. The victim will respond back with a TCP packet with SYN and ACK flags set. In this process the victim allocates system resources, waiting for a complete handshake. With too many half-open connections, most systems crash. CERT has a very good definition of SYN flood attacks, that can be found at http://www.cert.org/advisories/CA-1996-21.html.

In each box it is executed the following commands:

```
# juno-z101f   [ns (1s/10^9) delay] [threads (dfl:1)]
```

The IP address (a.b.c.d) used will be the IP addres of GIAC's web server, with no delay and 5 threads, maximizing the attack.

```
# juno-z101f a.b.c.d 80 0 5
juno-z.c by Sorcerer
target=a.b.c.d:80 delay=0
using 5 threads, pids: 19453(main) 19457 19456 19455 19454
```

4.2.2 - Dealing with the DDoS problem

Today the Distributed Denial of Service presents a very serious problem to the Internet. This kind of attack generally depends on a large number of compromised machines and it is almost imposible to fix all machines connected. To deal with this problem it is necessary a distributed solution, using a single protection doesn't mean we're safe. In most cases the ISP must be contacted to help tracking and solving the problem and it's not unusual the ISP contact another ISP in this process.

This problem have been observed and studied for years, and since then we have only a few tools to deal with this problem. The IETF - http:/www.ietf.org Internet Area working group is proposing the ICMP Traceback (or itrace) to deal with certain denial of service atracks using forged source IP addresses. There are other proposals such Pi (Path Identifier) and Pushback. The first one is an academic proposal without real implementation. Commercial tools are also available, such as PowerSecure by Mazu Networks - http://www.mazunetworks.com, PeakFlow by Arbor Networks - http://www.arbornetworks.com, FloodGuard by NetZentry - http://www.netzentry.com, among others.

4.3 - Attacking an Internal Server

To attack the internal server of XYZ design we will compromise the workstation of a teleworker using Sub7. First we send SPAM with the infected sub7 executable to all known valid e-mail addresses of GIAC employees. It is assumed we have lots of information of such employees using social engineering techniques before doing anything.

The SPAM would be very appealing, forging the security officer email informing to all users to install the updated version of the VPN client. We hope the teleworker isn't using a personal firewall and anti-virus.

Hopefuly at least one teleworker ``upgraded'' the VPN client and get infected with Sub7. After being able to connect to the Sub7 server, we install a key logger in the system, that will capture every keystroke. The keylogger will send to a anonymous e-mail account all log generated every day in case we're not able to connect to Sub7 server anymore.

Now we have found the VPN connection uses just a username and password. All necessary information was collected by the keylogger installed on teleworker's machine.

4.3.1 - Countermeasures

- Security Policy enforcement - The security policy must include that is mandatory the installation of an anti-virus and personal firewall on all remote workstation. This minimize (but not solve) the problem of being hacked and then then compromise the internal server.
- The network layout must be layered - Internal Users and RoadWarriors must be placed in a separated subne then compromise the internal server.
- The network layout must be layered - Internal Users and RoadWarriors must be placed in a separated subnet, protected by a firewall. The same should be done with all internal servers.
- The VPN gateway should use only X509 digital certificates. Using pre-shared key or username/password should not be used by remote users.
- Education - All employees must be warned about the social engineering techniques, to avoid being fooled by a smart guy posing as a System Administrator or the CEO. Social engineering attacks are very common form of hacking, and most cases, has a very successul rate.

Links and References

INTERNET PROTOCOL V4 ADDRESS SPACE (last updated 2003-04-05)

URL:http://www.iana.org/assignments/ipv4-address-space

Address Allocation for Private Internets

URL:http://www.ietf.org/rfc/rfc1918.txt?number=1918

Congestion Control in IP/TCP Internetworks

URL:http://www.ietf.org/rfc/rfc0896.txt

ICMP Packet Filtering v1.2

URL:http://www.cymru.com/Documents/icmp-messages.html

Secure IOS Template Version 3.0 08 APR 2003

URL:http://www.cymru.com/Documents/secure-ios-template.html

Linux 2.4 Packet Filtering HOWTO $Revision: 1.26

URL:http://www.netfilter.org/documentation/HOWTO//packet-filtering-HOWTO.html

Life with qmail - 16 August 2003

URL:http://www.lifewithqmail.org/lwq.html

CERT Advisory CA-1996-21 TCP SYN Flooding and IP Spoofing Attacks

URL:http://www.cert.org/advisories/CA-1996-21.html

Snort "Snort: The Open Source Network Intrusion Detection System" 2002

URL:http://www.snort.org

NTP Time Synchronization Server

URL:http:/www.ntp.org

Challenges and Principles of DDoS Defense

URL:http://www.cs.ucla.edu/~sunshine/publications/defcom-sig.pdf

A Taxonomy of DDoS Attacks and Defense Mechanisms

URL:http://www.cs.ucla.edu/~sunshine/publications/ccr.pdf

FreeS/WAN 2.02 documentation

URL:http://www.freeswan.org/freeswan_trees/freeswan-2.02/doc/index.html

FreeS/WAN IPSec Interoperability Guide

URL:http://www.ssh.com/documents/31/ssh_sentinel_14_freeswan.pdf

Red Hat Network Quick Start Guide

URL:https://rhn.redhat.com/help/quickstart.pxt

Bugtraq vulnerability list

URL:http://www.securityfocus.com/archive/1

Brad Tauer GCFW Practical assignment

URL:http://www.giac.org/practical/GCFW/Brad_Tauer.pdf

Lin Zhu GCFW Practical assignment

URL:http://www.giac.org/practical/GCFW/Lin_Zhu_GCFW.pdf

Eeye's Retina Network Scanner

URL:http://www.eeye.com