Global Information Assurance Certification Paper

Practical

GCFW
ver 1.8
SANSFIRE 2002 BOSTON, MA

Sven Olensky

Preface

This documentation represents my attempt to achieve the GCFW/ GIAC Certified Firewall Analyst Certification. Four assignments were required to be worked and completed.

Many, many special thanks to my wife who endured me during the days and weeks of writing this documentation. I am not sure how she did it, but she did, and I could complete this Practical. Thank you, I really love you. Du bist das Groesste fuer mich auf dieser Welt.

Thanks also to my dogs, who once in a while distracted me and forced me to get away from the laptop for a little bit, onto the fresh air.

Thanks also to Bernd, who helped me filtering out some good suggestions for networks to be tested in the 4th assignment. He saved me quite some time.

Sven Olensky
December 2002
Atlanta, GA, USA

Note: this document was prepared in Word DOC format. The converted PDF version may experience loss in quality (e.g. screenshots of firewall rules, diagrams etc may not be legible after conversion). Try to get in touch with me if you need the DOC version of this document.

Contents

0. Introduction 7
1. Assignment 1 - Security / Network Architecture 8
1.1. General Architecture 9
1.1.1. diagram 9
1.1.2. Security Perimeter / Layers of Defense 9
1.1.3. Server/Device features and Layers of Security 10
1.1.3.1. Overview 10
1.1.3.2. Server Details - Summary 10
1.1.3.3. Border Router 12
1.1.3.4. Outside Firewall Cluster and Inside Firewall Cluster 12
1.1.3.5. Firewall Management Server 13
1.1.3.6. High Availability/ Redundancy 13
1.1.3.7. IDS 14
1.1.3.8. Syslog 14
1.1.3.9. SecurID 14
1.1.3.10. Jumpstart 15
1.1.3.11. Tape/Backup 15
1.1.3.12. Protected Internal Mail 15
1.1.3.13. Split DNS 15
1.1.3.14. Web Server 15
1.1.3.15. FTP Server 15
1.1.3.16. Database Servers 15
1.1.3.17. Terminalserver 16
1.1.3.18. Host-based firewalls 16
1.1.3.19. NTP 16
1.1.3.20. Server Hardening 16
1.2. Cabling / Layer 1 17
1.2.1.1. diagram 17
1.2.1.2. description 17
1.3. Data link layer / Layer 2 19
1.3.1. diagram 19
1.3.2. description 20
1.4. Network Layer / Layer 3 21
1.4.1. diagram 21
1.4.2. description 22
1.4.3. Routing 23
1.4.4. IP Address Sheet 24
1.5. Required Skillsets 28
1.6. Access requirements/restrictions 29
1.6.1. Customers 29
1.6.2. Suppliers and Resellers 29
1.6.3. GIAC Enterprises employees located on GIAC Enterprise's internal network 29
1.6.4. GIAC Enterprises mobile sales force and teleworkers 29
1.6.5. Employees Access to the outside world 30
1.6.6. Additional Access Requirements 30
2. Assignment 2 – Security Policies 31
2.1. Border Router 32
2.1.1. Configuration 32
2.1.1.1. general configuration on the border router 32
2.1.1.2. Inbound access lists 33

2.1.1.3. Outbound access lists 34
2.1.1.4. SecurID support on the Border Router 36
2.1.1.5. syslogging and SecurID authentication traffic 37
2.1.2. Rule Order 38
2.2. VPN / core firewall policies 39
2.2.1. Introduction 39
2.2.2. VPN details 39
2.2.3. Split Tunneling / VPN client side 39
2.2.4. Explanation of the Naming Conventions used in the Policies 40
2.2.4.1. Groups 40
2.2.4.2. Nodes 40
2.2.5. Outside Firewall Policy 41
2.2.5.1. Rule 1 41
2.2.5.2. Rule 2 41
2.2.5.3. Rule 3 41
2.2.5.4. Rule 4 41
2.2.5.5. Rule 5 41
2.2.5.6. Rules 6 through 10 42
2.2.5.7. Rules 11 through 17 42
2.2.5.8. Rule 18 42
2.2.5.9. Rule 19 42
2.2.5.10. Rules 20 and 21 43
2.2.5.11. Rule 22 43
2.2.5.12. Rule 23 43
2.2.5.13. Rule 24 43
2.2.5.14. Rule 25 43
2.2.5.15. Rule 26 43
2.2.5.16. Rule 27 44
2.2.5.17. Rule 28 44
2.2.5.18. Rule 29 44
2.2.5.19. Rule 30 44
2.2.5.20. Rule 31 44
2.2.5.21. Rule 32 44
2.2.5.22. Rule 33 45
2.2.5.23. Rule 34 45
2.2.5.24. Rule 35 45
2.2.5.25. Rule 36 45
2.2.5.26. Rule 37 45
2.2.5.27. Rule 38 45
2.2.6. Inside Firewall Policy 46
2.2.6.1. Rule 1 46
2.2.6.2. Rule 2 46
2.2.6.3. Rule 3 46
2.2.6.4. Rule 4 46
2.2.6.5. Rules 5 through 11 46
2.2.6.6. Rule 12 47
2.2.6.7. Rule 13 47
2.2.6.8. Rule 14 47
2.2.6.9. Rule 15 47
2.2.6.10. Rule 16 47
2.2.6.11. Rule 17 48
2.2.6.12. Rule 18 48
2.2.6.13. Rule 19 48

2.2.6.14. Rule 20 48
2.2.6.15. Rule 21 48
2.2.6.16. Rule 22 48
2.2.6.17. Rule 23 49
2.2.6.18. Rule 24 49
2.2.6.19. Rule 25 49
2.2.6.20. Rule 26 49
2.2.6.21. Rule 27 49
2.2.7. NATting 50
2.2.7.1. Proxy ARP 50
2.2.8. Rule order 52
2.3. Tutorial 52
2.3.1. Rule Syntax 52
2.3.2. How to create a Rulebase from scratch (in NG) 54
2.3.3. Installation of a Policy and Example for Troubleshooting 58
2.3.4. Verifying the Rules in the Firewall Logs 62
3. Assignment 3 - Audit 63
3.1. Planning the Audit 64
3.1.1. Technical Approach 64
3.1.2. Time of Day for doing the Audit 65
3.1.2.1. Day of Week 65
3.1.3. Costs and Level of Effort 65
3.1.3.1. Preparations 65
3.1.4. Risks and Considerations 66
3.1.4.1. Backup procedures before the Audit 66
3.2. Conducting the Audit 66
3.2.1. Conventions 66
3.2.2. Firewall Configuration Audit 66
3.2.3. Scan from the Outside 67
3.2.4. Rule 1 70
3.2.5. Rule 2 70
3.2.6. Rule 3 71
3.2.7. Rule 4 73
3.2.8. Rule 5 74
3.2.9. Rules 6 through 10 75
3.2.10. Rules 11 through 17 78
3.2.11. Rule 14 78
3.2.12. Rule 18 80
3.2.13. Rule 19 81
3.2.14. Rule 20 and 21 81
3.2.15. Rule 22 82
3.2.16. Rule 23 82
3.2.17. Rule 24 83
3.2.18. Rule 25 84
3.2.19. Rule 26 85
3.2.20. Rule 27 85
3.2.21. Rule 28 86
3.2.22. Rule 29 86
3.2.23. Rule 30 87
3.2.24. Rule 31 88
3.2.25. Rule 32 88
3.2.26. Rule 33 89
3.2.27. Rule 34 90

3.2.28. Rule 35 90
3.2.29. Rule 36 90
3.2.30. Rule 37 91
3.2.30.1. Conventions 91
3.2.30.2. the Audit 91
3.2.31. Rule 38 92
3.3. Evaluation 93
3.4. Analysis and Recommendations 93
3.4.1. Improvements / Corrections in the Firewall Policy 93
3.4.2. Improvements to the Network Architecture 96
3.4.2.1. Alternative Network Architecture 97
3.4.2.2. Cost considerations 97
3.4.3. Other Suggestions 98
4. Assignment 4 – Design Under Fire 99
4.1. Chosen Practical 100
4.2. Attack against the Firewall itself 101
4.2.1. Designing the Attack 102
4.2.1.1. Pseudocode for the Exploit 102
4.2.2. Execution of Attack 104
4.2.3. Results of the Attack 105
4.2.4. Countermeasures To Vulnerability 105
4.3. Denial of Service Attack 105
4.3.1. Tribal Flood Network 2K 105
4.3.1.1. Description 105
4.3.1.2. Usage 107
4.3.2. Compromising Machines that will Execute the Attack 107
4.3.3. The Attack 108
4.3.4. Countermeasures To DDoS Attack 109
4.4. Compromise an Internal System 109
4.4.1. Target Selection 109
4.4.2. Executing the attack 111
4.4.3. Countermeasures 112
4.4.4. Will we be noticed? 112
5. References 114

0. Introduction

GIAC Enterprises, an e-businees which main revenue generator is the online sale of fortune cookie sayings, desire to build a new network that better fits their needs, dubbed the "eKookie network". In the first couple of months after the initial launch of the e-cookie trade, servers, network devices and connectivity to the Internet were primarily borrowed from Corporate, since the eKookie branch was merely an experiment in the first place – a successful experiment as people realized as soon as they saw the immense interest that got spawned by the new eKookie website, www.ekookie.com. The immense traffic generated by the website soon used up the small corporate network bandwidth, so IT is forced to deploy a new network as soon as possible.

To achieve this, GIAC Enterprises has contracted out the initial network design and implementation to OSec Inc. which is owned by me (OSec=Olensky Security Network & Design Inc.). The goal is to deploy a secure, fast and cost-efficient network architecture that meets todays requirements and demands, but is also reliable and should be able to withstand the "demands of tomorrow". In order to finance this, GIAC has gotten assigned a sufficient budget from GIAC Holdings, the parent company that also owns some of the suppliers and resellers GIAC Enterprises will interact with.

GIAC Enterprises wants to be the biggest Internet retailer for fortune cookie sayings. In order to do this, they demand a network that can meet even the highest service level agreements and is almost always available for business to take place. GIAC already has contracts with Cisco, Nokia and Sun, so they will get discounts when they purchase (a lot) of equipment.

1. Assignment 1 - Security / Network Architecture

The requirements of GIAC Enterprises stated:

Define a network security architecture for GIAC Enterprises, an e-business which deals in the online sale of fortune cookie sayings.

Your architecture must consider access requirements (and restrictions) for:
- Customers (Companies or individuals that purchase bulk online fortunes)
- Suppliers (Companies that supply GIAC Enterprises with their fortune cookie sayings)
- Partners (International companies that translate and resell fortunes)
- GIAC Enterprises employees located on GIAC Enterprise's internal network
- GIAC Enterprises mobile sales force and teleworkers

You must explicitly define how the business operations of GIAC Enterprises will take place.
- How will each of the groups listed above connect to or communicate with GIAC Enterprises?
- How will GIAC Enterprises employees access the outside world? What services, protocols, or applications will be used?

Defining access requirements and the reasoning for those requirements is critical to this assignment. [..]

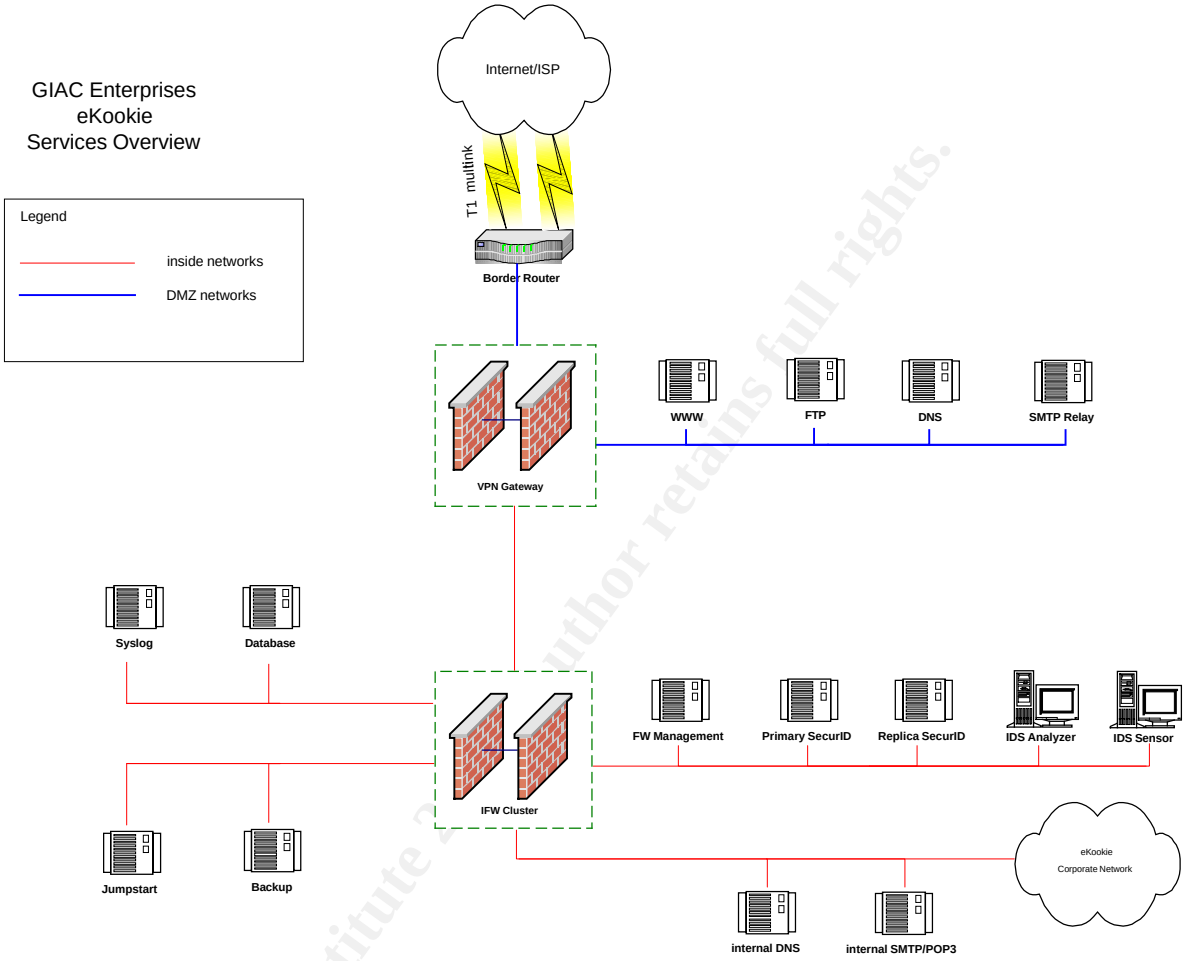
In designing your architecture, you must include the following components:
- Filtering Router(s)
- Firewall(s)
- VPN(s)
- An IP addressing scheme (use known non-routable addresses your policies will be graded taking them into account.)
- Your architecture may also include the following optional components if they are appropriate to your design:
    - Internal firewalls (Are internal firewalls appropriate for additional layered protection; to segment internal networks…?)
    - Additional secure remote access (Is additional remote access – other than the VPN – required by administrators, salespeople, telecommuters…?).
    - Intrusion detection systems

You must include a diagram or set of diagrams that shows the layout of GIAC Enterprise's network and the location of each component listed above. You must provide the specific brand and version of each perimeter defense component used in your design. Finally, include an explanation that describes the purpose of each component, the security function or role it carries out, and how the placement of each component on the network allows it to fulfill this role. The network can be as complex or as simple as you like as long as it meets the functional requirements that you define according to the guidelines given above. The important thing is not how elaborate your network is, but that your design actually works.

You must justify the appropriateness of your design. Is it both technically reasonable and financially feasible? Are you building a $1000 fence to contain a $100 horse? You may provide a cost or bill of materials if you wish.

1.1. General Architecture

1.1.1. diagram



1.1.2. Security Perimeter / Layers of Defense

The security perimeter is setup like this:

- the first line of defense is the border router. Very specific access-lists allow only needed traffic through to the network, everything else gets dropped and logged
- the second line of defense is the outside firewall cluster that also serves as VPN endpoint for site-to-site VPNs and VPN clients. It only allows traffic to the networks as needed, and drops everything else.
- another layer of defense (third line) is the inside firewall cluster: it is an additional shield for the internal networks (internal services and internal corporate network)
- the fourth line of defense are hostbased firewalls: there is a hostbased firewall installed on every server that only allows needed ports and drops everything else and logs it to the syslog facility

- additional protection is given by the internal syslog facility: all servers and devices log system-level and auth-level events to this server. The logs are monitored; upon certain events they are used for informing/alerting staff etc.
- another line of defense (or better: alerting mechanism) is the IDS facility: it watches public traffic and alerts on suspicious behaviour (to the syslog facility)

1.1.3. Server/Device features and Layers of Security

1.1.3.1. Overview

The equipment used are:

- a Cisco 3640 Router that faces the Internet / serves as border router
- 2 Cisco 4006 switches, outside and inside switch, modular, Gbit capable
- 2 pairs of Nokia IP 530 firewalls with Checkpoint FW-1 Next Generation
- 11 Sun Netra X1 machines and 2 Sun Enterprise 280R servers. All are running Solaris 8 servers, for the services. The 280Rs will be used for the database and the backup servers. The Netras were deemed sufficient. All servers will be equipped with quad cards to facilitate Sun Multipathing (needs 4 ports). In addition, the internal ethernet port will be used for management.
- two NetBSD based PCs for the IDS work, one of them will serve as IDS sensor and will be connected to the SPAN port of the core switch, the other NetBSD machine will serve as analyzer to parse and summarize the scan results
- one IN-REACH Itouch Terminalserver

1.1.3.2. Server Details - Summary

This list contains a summary of the servers, used hardware and the OS version running on them

Server	Hostname(s)	Type	additional hardware	OS/version/software
Network Devices				
Router	ROUTER	Cisco 3640	2 serial, 2 ethernet interfaces	IOS 12.2T
outside switch	OSW01	Cisco 4006 Catalyst		IOS 12.2T
inside switch	ISW01	Cisco 4006 Catalyst		IOS 12.2T
OFWCluster	OFW01 and OFW02	Nokia IP530	1 quad card, 1 VPN accelerator each	IPSO 3.5FCS10 Checkpoint FW-1 NGFP3
IFWCluster	IFW01 and IFW02	Nokia IP530	2 quad cards each	IPSO 3.5FCS10 Checkpoint FW-1 NGFP3
Terminalserver	TSERV01	INREACH-Itouch	flash card	In-Reach Level 4, V3.0S13

Servers **(latest versions of software packages and patchclusters for the Solaris systems)**

WWW	WWW01	Netra X1	2x18 GB HDD, 512 MB RAM	Solaris 8 Apache 2.0.43 or later
FTP	FTP01	Netra X1	2x18 GB HDD, 512 MB RAM	Solaris 8 ProFTPd 1.2.26 or later
external DNS	NS01	Netra X1	2x18 GB HDD, 512 MB RAM	Solaris 8 BIND 9.2.1
internal DNS	NSI01	Netra X1	2x18 GB HDD, 512 MB RAM	Solaris 8 BIND 9.2.1 or later
Syslog	SYSLOG01	Netra X1	2x18 GB HDD, 512 MB RAM	Solaris 8
Database	DB01	Enterprise 280R	2x36 GB HDD, 1024 MB RAM	Solaris 8 Oracle 9i or later
Jumpstart	JUMP01	Netra X1	2x36 GB HDD, 512 MB RAM	Solaris 8
Tape Backup	TAPE01	Enterprise 280R	2x36 GB HDD, 1024 MB RAM	Solaris 8
Mail Relay	MX01	Netra X1	2x18 GB HDD, 512 MB RAM	Solaris 8 sendmail 8.12.6 or later
internal Mail/Pop3	mail.corp / pop.corp	Netra X1	2x18 GB HDD, 512 MB RAM	Solaris 8 sendmail 8.12.6
FW management	FWMGMT01	Netra X1	2x18 GB HDD, 512 MB RAM	Solaris 8 CP FW NG FP3
Primary SecurID	SECURID01	Netra X1	2x18 GB HDD, 512 MB RAM	Solaris 8 RSA ACE/server 5.0.3
Replica SecurID	SECURID02	Netra X1	2x18 GB HDD, 512 MB RAM	Solaris 8 RSA ACE/server 5.0.3
IDS Analyzer	IDSANALYZER01	AMD Athlon XP-2100	1x60 GB IDE HDD, 256 MB RAM	current Free/NetBSD snortsnarf etc
IDS Sensor	IDSSENSOR01	AMD Athlon XP-2100	2x80 GB IDE HDD, 512 MB RAM	current Free/NetBSD snort

1.1.3.3. Border Router

The border router (Cisco 3640 running IOS 12.2T) has many basic protection features enabled: it blocks internal traffic from hitting the outside, it has many unneeded services disabled (finger, echo etc) and it only allows certain traffic to hit certain public servers in the network. A detailed configuration overview can be seen later in this document.

The router will have 4 interfaces: 2 serial interfaces grouped in a Multilink interface and 2 FastEthernet interfaces. 1 FastEthernet is for the Internet-GIAC production traffic, the other one serves as out of band management interface, going into the external device management network (managed by the outside firewalls as well).

1.1.3.4. Outside Firewall Cluster and Inside Firewall Cluster

The inside networks are shielded from the Internet by a dedicated (inside) pair of firewalls (all firewalls are Nokia IP530 appliances running the latest stable version of IPSO OS): public traffic to the DNS, Webserver and the mail relay does not touch that inside firewall cluster. The outside firewall cluster is responsible for protecting the public servers, as well as serving as VPN gateway for external employees, so that they will be able to reach internal resources, as well as VPN endpoint for vendors etc that desire to build site-to-site VPNs with GIAC Enterprises.

It was determined that the firewalls will be equipped with enough quad-100Mbit-network cards to facilitate the demand on network connections. In addition, the outside firewalls will be equipped with VPN accelerators to take the encryption load off the processing power of the firewalls themselves.

1.1.3.4.1 Interface Assignments

Derived from the network architecture design that follows later in this section, the following ports on the firewalls were assigned to each network

outside firewalls (1 quad card each)

eth-s1p1c0	**router to firewall VLAN**
eth-s1p2c0	**public server VLAN**
eth-s1p3c0	**public server mgmt VLAN**
eth-s1p4c0	**OFW to IFW VLAN**
eth1c0	**state sync OFW VLAN**
eth2c0	**external network dev. mgmt VLAN**
eth3c0	unused
eth4c0	unused

inside firewalls (2 quad cards each)

eth-s1p1c0	**OFW to IFW VLAN**
eth-s1p2c0	**DB and Syslog service VLAN**
eth-s1p3c0	**Security VLAN**
eth-s1p4c0	**JUMPSTART, Backup service VLAN**
eth-s2p1c0	**DB and Syslog mgmt VLAN**
eth-s2p2c0	**Security mgmt VLAN**
eth-s2p3c0	**JUMPSTART, TAPE mgmt VLAN**
eth-s2p4c0	**Corporate network connection**
eth1c0	**corporate server mgmt VLAN**
eth2c0	**state sync IFW VLAN**
eth3c0	unused
eth4c0	unused

1.1.3.4.2 SmartDefense

We will be using SmartDefense from Checkpoint on our outside firewall cluster. That way, DDoS/DoS attacks may be mitigitated. Of course, you can always overwhelm a network, it only depends on how many clients/agents are attacking the firewall/network at the same time. However, this may mitigate most "common" DOS attacks that happen on day-to-day basis. More info here: [SMARTDEFENSE].

1.1.3.5. Firewall Management Server

The management of the firewalls happen from the inside network (this server is a Netra X1 running Checkpoint NG FP3 on Solaris 8). Administrators are only able to connect with the GUI-clients (policy editor, log viewer) if they use the VPN client, for security reasons. Access from the corporate network without using the VPN client may be considered later on when it can be restricted down to one or two specific machines the security administrators will be using.

1.1.3.6. High Availability/ Redundancy

Both firewall-pairs are clustered using VRRP [VRRP]. The outside firewalls, inside firewalls are state synced within the pair and provide for HA. Everything is setup in active-passive configurations, so the passive partner in the pair waits until the active partner fails/ stops

responding etc, then it takes over. Established sessions in the firewalls will be kept up since every active session/connection is copied into the state table of the passive partner.

All production relevant servers (DNS, DB, FTP, WWW, SYSLOG, mail servers, SecurID servers) are utilizing Solaris Multipathing [MULTIPATHING]. For this to work, every server needs 5 assigned IP addresses and 2 separate subnet connections: one subnet contains the live traffic, the other subnet serves for testing of the active interfaces The "service IP" is a "floating IP" that gets managed by the active interface. There are 4 ports for this in any of those servers; if the testing for the active ports fail, the system switches the network configuration to the other ports.

All production relevant devices have additional power supplies.

1.1.3.7. IDS

The network has two machines that are related to intrusion detection: the IDS sensor and the IDS analyzer. The machines are PCs running the latest stable-FreeBSD. The IDS sensor is running Snort (latest stable release). The sensor is connected to a SPAN port in the outside switch and is actively listening to any incoming traffic from the Internet that was passed through the router (after filtering it through the access-lists on the router). Since the SPAN port is not a "real" interface (i.e. traffic cannot be actually passed through it and responded to), it is not considered a risk to place the "real" network connections (management and service networks) for this IDS sensor behind the internal firewall cluster. Both service and management network connection is shared with the SecurID and the firewall management servers, as well as the IDS analyzer machine. In preset intervals (every 8 hours or so), the analyzer connects to the IDS sensor, copies the scanner logs off the sensor into its analyzer-directory, bounces the snort process (to reinitialize the log files) and runs an analyzer (like SnortSnarf) over the log files. The files that were generated will then get emailed to the security administrators. The finetuning of the IDS is an ongoing process; some events will get sent to the syslog facility on the SYSLOG logging server and admins will be alerted.

1.1.3.8. Syslog

The network has a dedicated syslog facility (the server is a Netra X1 running Solaris 8). All servers and devices in the network generate syslog messages of varying severity, a certain set (e.g. all authentication messages) gets forwarded to the SYSLOG01- logserver for further processing. A log watcher (swatch) is running on the log server to take action on certain events. For example, it sends out emails to the admins if 'su root' attempts fail, unknown users try to log into a server, servers hit their capacity etc etc. The logfiles get rotated and archived in pre-determined intervals.

1.1.3.9. SecurID

All servers, network devices and VPN connections are protected by SecurID, a two-token based authentication mechanism that avoids the use of one-time passwords. Everybody accessing shells on any server or using the VPN client to access the inside networks from the Internet has to use his/her SecurID fob in order to be able to authenticate. The matter of who can access what devices/servers is configured a) on the ACE/server (SecurID server) side and b) in the VPN user setup (users belonging to certain groups can access certain resources when they are using the VPN client).[RSA]

The servers are Netra X1s running Solaris 8.

1.1.3.10. Jumpstart

The Jumpstart server contains readily available images for every server type (e.g preinstalled DNS server including up-to-date patches and configuration) in order to be able to quickly rebuild a server if needed.

1.1.3.11. Tape/Backup

All servers and devices get backed up. The initial backup is a complete backup, all further backups are done incrementally, i.e. only changes are backed up. Every once in a while (e.g. every month), all servers will also get backed up fully and the old backups get rotated out on tape. The backup system is using Veritas Netbackup. The server is a Sun 280R running Solaris 8.

Together with the Jumpstart server, it is possible to completely rebuild an existing server from scratch.

1.1.3.12. Protected Internal Mail

All corporate email from the Internet hits the primary mail relay in the public DMZ network, then gets forwarded to the internal mail server. The only SMTP communication allowed is Internet -> Mail Relay -> internal mail server and internal mail server -> Internet (with the internal mail server being NATted to its outside address)

1.1.3.13. Split DNS

The network architecture features separated external and internal DNS servers (both servers are Netra X1s running Solaris 8): external DNS for resolving public IPs, internal DNS for resolving internal IPs within the network (to the inside), as well as caching responses that got retrieved from the external DNS and the Internet. The internal DNS acts as primary DNS for the inside, and as cache/forwarder for all other requests. A backup (secondary) DNS for the public side will be provided for by the ISP. The internal DNS will start querying that secondary server in case the primary does not respond. No other direct communication between the internal DNS and the Internet will be permitted. Both the internal DNS and the internal mail server have a static NAT on the inside firewall and the communication to those internal servers will be as restricted as possible.

1.1.3.14. Web Server

The web server (Netra X1 running Solaris 8) uses the latest stable release of Apache with modssl, to offer both HTTP and HTTPS – based transactions.

1.1.3.15. FTP Server

The FTP server (Netra X1 running Solaris 8) is only accessible through site-to-site VPNs. The FTP protocol offers many risks (clear text etc) that need to get contained, thus the site-to-site VPNs. It was thought about using SSH for transactions between suppliers, resellers and GIAC, but it was decided to use FTP instead. Some resellers use specific software for the transactions (processing etc) that needed ftp. SFTP was another option, but was rejected for the same reason. The FTP server is running the latest stable version of ProFTPd and is also protected by a hostbased firewall, ipfilter (as all other boxes).

1.1.3.16. Database Servers

The heart of the eKookie business is the database where all the fortune cookie sayings are kept. Suppliers and Employees upload the cookie sayings to the FTP server, which in turn get inserted into the DB server's databases in regular, pre-defined intervals (scripts on the DB pull that information from the FTP server). The eShop Application running on the webserver pulls the needed information from the Database server (eShop being used as term for whatever

application handles the customer/webpage engine). The database server is a Enterprise 280R running Oracle 9i (or whatever the latest stable release is) on Solaris 8.

1.1.3.17. Terminalserver

There will be a terminalserver in this network to facilitate remote administration of the console ports of the equipment. The console ports of the servers and devices are reachable via ssh to the high ports of the terminalserver (2122-6522 for ssh1). It utilizes RSA SecurID for authentication before access to the console ports is granted. The protocol used for communicating the SecurID information between the Terminalserver and the RSA ACE/servers (SecurID) is RADIUS.

1.1.3.18. Host-based firewalls

All servers are running host based firewalls (e.g. IPFilter) and are locked down. Only ports that are needed are opened up on the machines. Logging of dropped traffic goes to the SYSLOG server for further processing (alerting etc). The hostbased firewalls are managed by the security team.

1.1.3.19. NTP

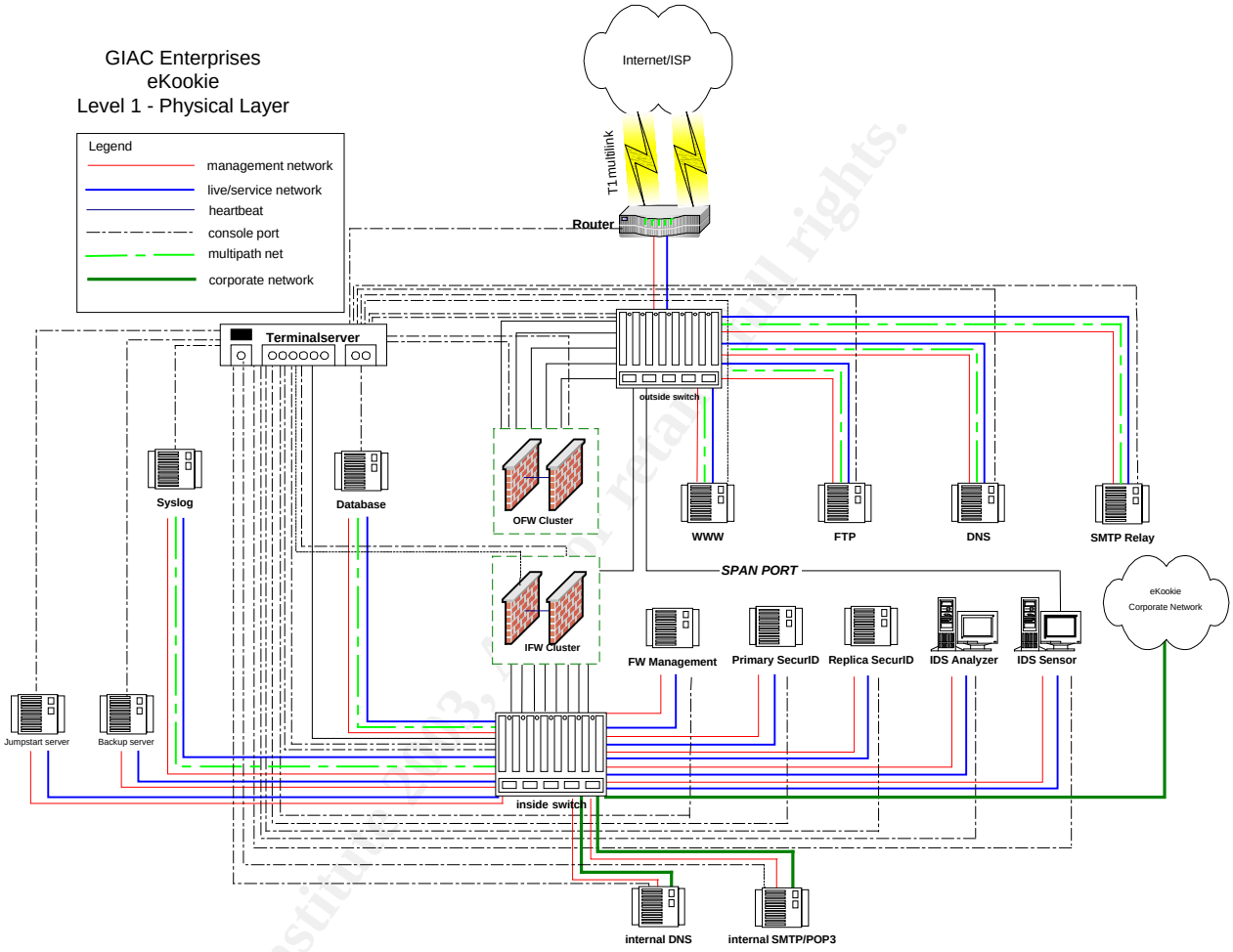
NTP is used to sync up the servers so that they are all on the same time. This is important for troubleshooting, logfile analysis etc. There will be two NTP servers in the network, running on the public DNS for outside servers needing to NTP sync and on the internal DNS, so that inside servers can timesync. The inside NTP will sync with the outside NTP, the outside NTP will sync with 3 dedicated official NTP servers in the Internet.

1.1.3.20. Server Hardening

All servers are hardened according to the most up-to-date standards (e.g using YASSP on the Sun machines [YASSP]), all ports are locked down except for the ones needed. Latest Sun patch clusters are installed.

1.2. Cabling / Layer 1

1.2.1.1. diagram



1.2.1.2. description

The bandwidth of the backbone is 100 MBit/sec. It was debated whether GIAC maybe wants to implement a Gbit backbone, but it was decided that 100Mbit should be enough for now, as long as the traffic going across that backbone is as separated as possible. This proposal will contain equipment that is Gbit capable, as it is likely that this will be needed in the near future, if the eKookie project grows as expected.

In general: all 100Mbit connections consist of category 5 unshielded twisted pair patch cables.

- the servers will be connected to the switches with 100Mbit cat 5 patch cables

- the firewall clusters will be interconnected for state syncing, with 100Mbit crossover cables, directly (no switch inbetween). The connections to the switches will also consist of 100Mbit Category 5 twisted pair patch cables.

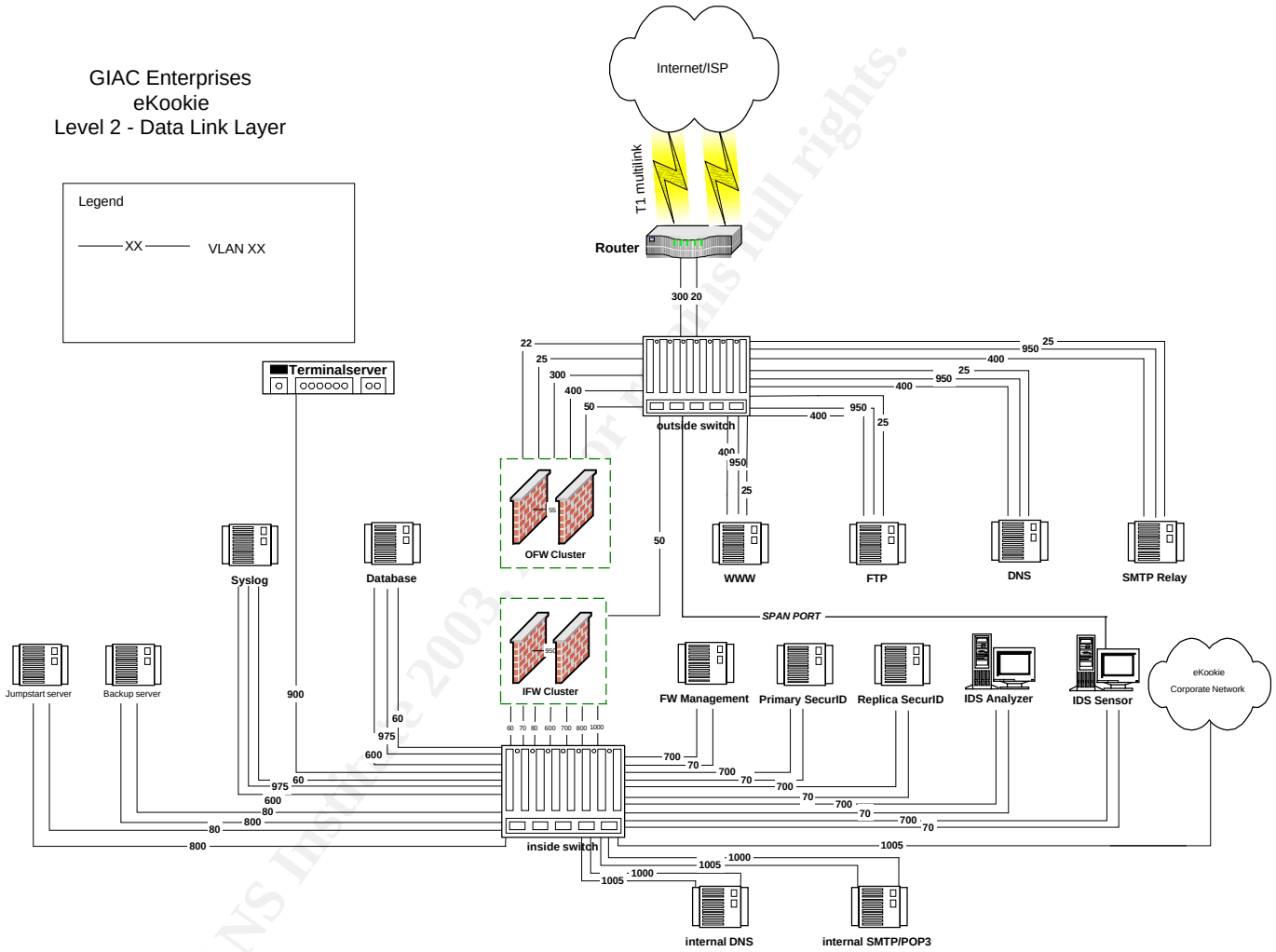
- There will be no physical connections between the outside and the inside switch for security reasons

- the terminalserver will have a management network connection of 100Mbit. This connection will also be used to access the server consoles (e.g. via ssh to the terminalserver ports to access the server consoles). The console cabling of the servers/devices depend on the type of device that is being accessed; e.g. Sun servers need different connectors than Cisco switches or Nokia firewall appliances

- the uplink to the Internet / ISP consists of two 1.5 MBit T1 connections that are bundled together in a so-called "multilink" which doubles the available bandwidth of the uplink to 3 MBit. 1.5MBit was deemed as not enough, a T3 connection (~42 MBit) was seen as not necessary (yet).

1.3. Data link layer / Layer 2

1.3.1. diagram



1.3.2. description

The networks are split up into several VLANs to minimize the amount of possible congestion. Public VLANs have publicly reachable IP addresses, private / management VLANs are internally/privately addressed with addresses from the RFC1918 networks. A detailed IP sheet can be seen later in this chapter. [RFC1918]

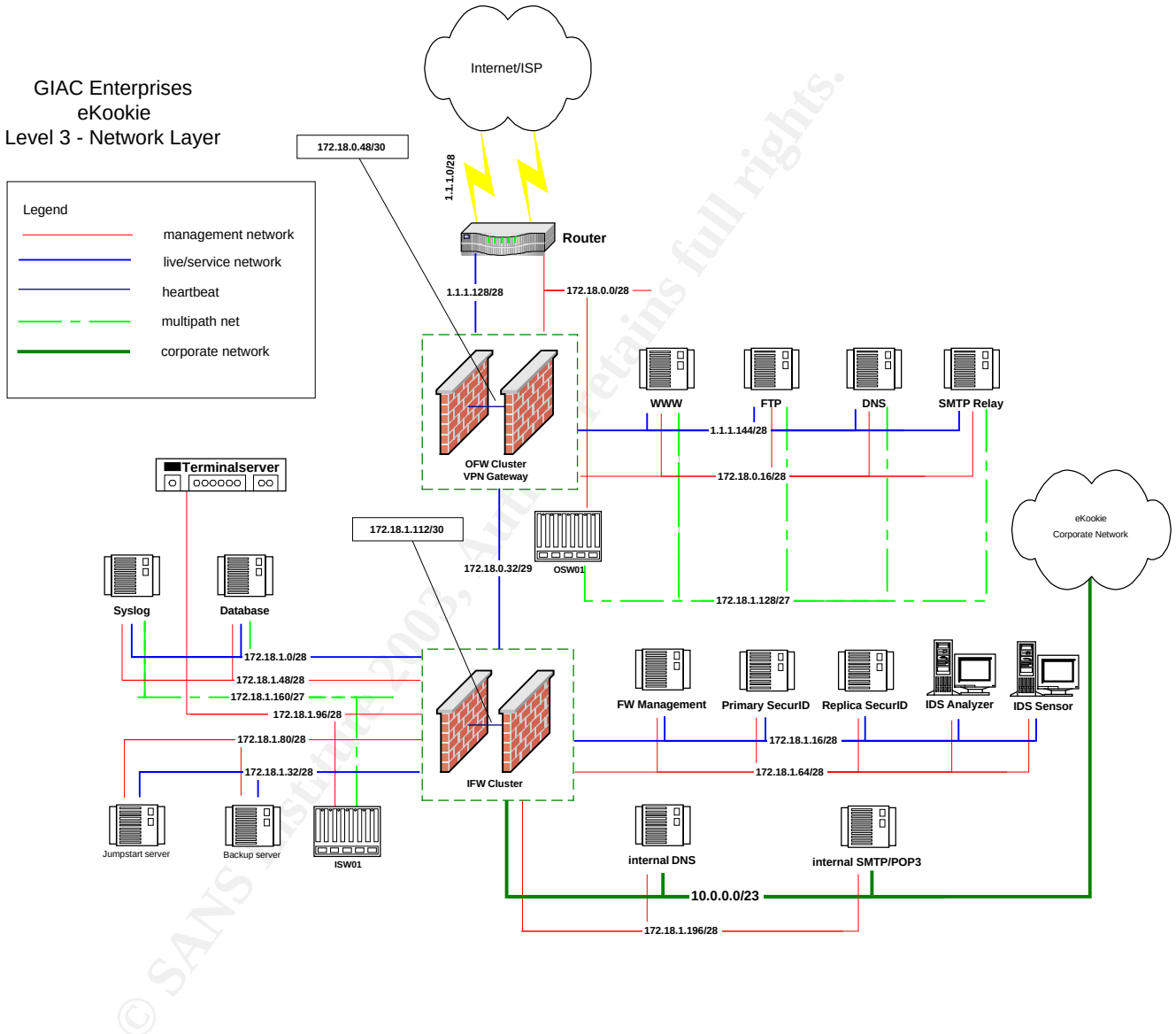
NOTE: the state-sync networks are also listed in here, for documentation and tracking purposes. They are not assigned/used on the switches, only used on the firewalls exclusively.

See below the list of VLANs and the description what each VLANs purpose is.

VLAN	Network	Description	Comments
20	**1.1.1.0/28**	**public traffic VLAN**	ISP to network connection
22	**1.1.1.128/28**	**router to firewall VLAN**	router to OFW connection
25	**1.1.1.144/28**	**public server VLAN**	public server network
50	**172.18.0.32/29**	**OFW to IFW VLAN**	OFW to IFW connection
55	**172.18.0.40/30**	**state sync OFW VLAN**	state sync for OFWs
59	**172.18.0.64/27**	**OFWCluster VPN POOL network**	IP Pool for OFWCluster VPN Client access
60	**172.18.1.0/28**	**DB and Syslog service VLAN**	DB/syslog production traffic
70	**172.18.1.16/28**	**Security VLAN**	FW mgmt/IDS Analyzer/SecurID production traffic
80	**172.18.1.32/28**	**Jumpstart, Backup service VLAN**	Jumpstart/Backup production traffic
300	**172.18.0.0/28**	**external network dev. mgmt VLAN**	mgmt for external network devices
400	**172.18.0.16/28**	**public server mgmt VLAN**	mgmt for external servers
600	**172.18.1.48/28**	**DB and Syslog mgmt VLAN**	mgmt for DB and Syslog servers
700	**172.18.1.64/28**	**Security mgmt VLAN**	mgmt for FW mgmt/IDS/SecurID servers
800	**172.18.1.80/28**	**Jumpstart, Backup mgmt VLAN**	mgmt for Jumpstart and Tape servers
900	**172.18.1.96/28**	**inside network dev mgmt VLAN**	mgmt for inside network devices
925	**172.18.1.112/30**	**state sync IFW VLAN**	state sync for IFWs
950	**172.18.1.128/27**	**outside servers multipath VLAN**	multipath test VLAN for outside servers
975	**172.18.1.160/27**	**inside servers multipath VLAN**	multipath test VLAN for inside servers
1000	**172.18.1.196/28**	**corporate server mgmt VLAN**	mgmt for corporate servers
1005	**10.0.0.0/23**	**corporate network connection**	connection to corporate switch

1.4. Network Layer / Layer 3

1.4.1. diagram



1.4.2. description

Following is the list of networks that will get routed by each device/cluster (networks each device/cluster is serving as default gateway for)

VLAN	Network	Description
Router		
20	**1.1.1.0/28**	**public traffic VLAN**
22	**1.1.1.128/28**	**router to firewall VLAN**
outside firewall cluster		
22	**1.1.1.128/28**	**router to firewall VLAN**
25	**1.1.1.144/28**	**public server VLAN**
50	**172.18.0.32/29**	**OFW to IFW VLAN**
55	**172.18.0.40/30**	**state sync OFW VLAN**
59	**172.18.0.64/27**	**Pool for OFWCluster VPN Client access**
300	**172.18.0.0/28**	**external network dev. mgmt VLAN**
400	**172.18.0.16/28**	**public server mgmt VLAN**
outside switch		
950	**172.18.1.128/27**	**outside servers multipath VLAN**
inside firewall cluster		
50	**172.18.0.32/29**	**OFW to IFW VLAN**
60	**172.18.1.0/28**	**DB and Syslog service VLAN**
70	**172.18.1.16/28**	**Security VLAN**
80	**172.18.1.32/28**	**Jumpstart, Backup service VLAN**
600	**172.18.1.48/28**	**DB and Syslog mgmt VLAN**
700	**172.18.1.64/28**	**Security mgmt VLAN**
800	**172.18.1.80/28**	**Jumpstart, Backup mgmt VLAN**
900	**172.18.1.96/28**	**inside network dev mgmt VLAN**
925	**172.18.1.112/30**	**state sync IFW VLAN**
1000	**172.18.1.196/28**	**corporate server mgmt VLAN**
1005	**10.0.0.0/23**	**corporate network connection**
inside switch		
975	**172.18.1.160/27**	**inside servers multipath VLAN**

The reason why the multipath VLANs are routed by the switches is that the traffic does not really need to get routed anywhere else.. rather the network only needs to "exist" for multipath testing of the interfaces to work right.

1.4.3. Routing

The public IP space was taken from IANA-reserved IP space [IANA].
The network 1.1.1.0/24 was assigned to the GIAC eKookie network.

The ISP/Uplink routes the 1.1.1.0/24 space to 1.1.1.1, the GIAC border router. The valid/used subnet with public servers is 1.1.1.128/25, which gets routed by the router to the firewall cluster (so, 1.1.1.128/25 via 1.1.1.130, the VRRP address of the OFW Cluster).

The firewalls will take care of the routing between the various subnetworks, based on security policies that are being enforced. The border router's only purpose is to do basic filtering of traffic coming into the network from the Internet and forwarding the benign traffic to the DMZ firewall which will then route that traffic to the proper destination subnetwork.

The routing on the servers depend on the networks they are in and whether they use multipathing or not. In general, the default routes on the servers point to the IP address of the VRRP address of the respective firewall cluster.

The default route of the inside firewall cluster is the VRRP address of the outside firewall cluster in VLAN 50, 172.18.0.34.

The default route of the outside firewall cluster is the IP of the border router in VLAN 22, 1.1.1.129.

The default route of the border router is the peer IP of the ISP, here 1.1.1.2.

1.4.4. IP Address Sheet

GIAC Enterprises
eKookie network
v 1.0, 12/01/02
Sven Olensky, OSEC Inc.

LEGEND

NODE-E0	publicly reachable IP address of node
NODE-Ex	other public IP address of node
NODE-M0	mgmt IP address of node
NODE-SIP	multipath service IP address of node
NODE-ETx	Test address no. X of node in multipath network
NODE-Hx	heartbeat address no. x of node in multipath network
NODE-VRRP	VRRP address of NODE-cluster
NODE-NAT	NATed address of internal node

Public IP's **1.1.1.0/24**

VLAN	Network	Description	IP address	Comments
20	**1.1.1.0/28**	**public traffic VLAN**	**1.1.1.1-14**	
		ROUTER-E0	1.1.1.1	
		ISP_ROUTER-E0	1.1.1.2	
22	**1.1.1.128/28**	**router to firewall VLAN**	**1.1.1.129-142**	
		ROUTER-VRRP	1.1.1.129	
		ROUTER-E1	1.1.1.130	
		ROUTER-E2	1.1.1.131	
		OFW VRRP-E0	1.1.1.132	
		OFW01-E0	1.1.1.133	
		OFW02-E0	1.1.1.134	
25	**1.1.1.144/28**	**public server VLAN**	**1.1.1.145-158**	
		OFW VRRP-E1	1.1.1.145	
		OFW01-E1	1.1.1.146	
		OFW02-E1	1.1.1.147	
		DNS01-SIP-E0	1.1.1.148	
		WWW01-SIP-E0	1.1.1.149	
		FTP01-SIP-E0	1.1.1.150	
		MX01-SIP-E0	1.1.1.151	
		CORP-NAT	1.1.1.152	
		INSIDEDNS01-NAT	1.1.1.153	
		INSIDEMX01-NAT	1.1.1.154	

Private IP's	172.18.0.0/23			

300	172.18.0.0/28	external network dev. mgmt VLAN	172.18.0.1-14	
		OFW VRRP	172.18.0.1	
		OFW01-M0	172.18.0.2	
		OFW02-M0	172.18.0.3	
		OSW01-M0	172.18.0.4	
		ROUTER-M0	172.18.0.6	

400	172.18.0.16/28	public server mgmt VLAN	172.18.0.17-30	
		OFW VRRP	172.18.0.17	
		OFW01	172.18.0.18	
		OFW2	172.18.0.19	
		DNS01-M0	172.18.0.20	
		WWW01-M0	172.18.0.21	
		FTP01-M0	172.18.0.22	
		MX01-M0	172.18.0.23	

50	172.18.0.32/29	OFW to IFW VLAN	172.18.0.33-38	
		OFW VRRP	172.18.0.33	
		OFW01	172.18.0.34	
		OFW02	172.18.0.35	
		IFW VRRP	172.18.0.36	
		IFW01	172.18.0.37	
		IFW02	172.18.0.38	

55	172.18.0.40/30	state sync OFW VLAN	172.18.0.41-42	
		OFW01	172.18.0.41	
		OFW02	172.18.0.42	

59	172.18.0.64/27	OFWCluster VPN IP POOL	172.18.0.65-94	

60	172.18.1.0/28	DB and Syslog service VLAN	172.18.1.1-14	
		IFW VRRP	172.18.1.1	
		IFW01	172.18.1.2	
		IFW02	172.18.1.3	
		DB01-SIP	172.18.1.4	
		SYSLOG01-SIP	172.18.1.5	

70	172.18.1.16/28	Security VLAN	172.18.1.17-30	
		IFW VRRP	172.18.1.17	
		IFW01	172.18.1.18	
		IFW02	172.18.1.19	
		FWMGMT01	172.18.1.20	
		SECURID01-SIP	172.18.1.21	
		SECURID02-SIP	172.18.1.22	
		IDSSENSOR01-SIP	172.18.1.23	
		IDSANALYZER01-SIP	172.18.1.24	

80	172.18.1.32/28	Jumpstart, Backup service VLAN	172.18.1.33-46	
		IFW VRRP	172.18.1.33	
		IFW01	172.18.1.34	
		IFW02	172.18.1.35	
		JUMP01-SIP	172.18.1.36	
		TAPE01-SIP	172.18.1.37	

600	172.18.1.48/28	DB and Syslog mgmt VLAN	172.18.1.49-62	
		IFW VRRP	172.18.1.49	
		IFW01	172.18.1.50	
		IFW02	172.18.1.51	
		DB01-M0	172.18.1.52	
		SYSLOG-M0	172.18.1.53	

700	172.18.1.64/28	Security mgmt VLAN	172.18.1.65-78	
		IFW VRRP	172.18.1.65	
		IFW01	172.18.1.66	
		IFW02	172.18.1.67	
		FWMGMT01-M0	172.18.1.68	
		SECURID01-M0	172.18.1.69	
		SECURID02-M0	172.18.1.70	
		IDSSENSOR-M0	172.18.1.71	
		IDSANALYZER-M0	172.18.1.72	

800	172.18.1.80/28	Jumpstart, Backup mgmt VLAN	172.18.1.81-94	
		IFW VRRP	172.18.1.81	
		IFW01	172.18.1.82	
		IFW02	172.18.1.83	
		JUMP01-M0	172.18.1.84	
		TAPE01-M0	172.18.1.85	

900	172.18.1.96/28	inside network dev mgmt VLAN	172.18.1.97-110	
		IFW VRRP	172.18.1.97	
		IFW01-M0	172.18.1.98	
		IFW02-M0	172.18.1.99	
		ISW01-M0	172.18.1.100	
		TSERVER01-M0	172.18.1.102	

925	172.18.1.112/30	state sync IFW VLAN	172.18.1.113-114	
		IFW01	172.18.1.113	
		IFW02	172.18.1.114	

950	172.18.1.128/27	outside servers multipath VLAN	172.18.1.129-158	
		OSW01	172.18.1.129	
		DNS01-H0	172.18.1.132	
		DNS02-H1	172.18.1.133	
		DNS01-ET0	172.18.1.134	
		DNS01-ET1	172.18.1.135	
		WWW01-H0	172.18.1.136	
		WWW01-H1	172.18.1.137	

		WWW01-ET0	172.18.1.138	
		WWW01-ET1	172.18.1.139	
		FTP01-H0	172.18.1.140	
		FTP01-H1	172.18.1.141	
		FTP01-ET0	172.18.1.142	
		FTP01-ET1	172.18.1.143	
		MX01-H0	172.18.1.144	
		MX01-H1	172.18.1.145	
		MX01-ET0	172.18.1.146	
		MX01-ET1	172.18.1.147	

975	**172.18.1.160/27**	**inside servers multipath VLAN**	**172.18.1.161-194**	
		ISW01	172.18.1.161	
		SYSLOG01-H0	172.18.1.164	
		SYSLOG01-H1	172.18.1.165	
		SYSLOG01-ET0	172.18.1.166	
		SYSLOG01-ET1	172.18.1.167	
		DB01-H0	172.18.1.168	
		DB01-H1	172.18.1.169	
		DB01-ET0	172.18.1.170	
		DB01-ET1	172.18.1.171	

1000	**172.18.1.196/28**	**corporate server mgmt VLAN**	**172.18.1.197-206**	
		IFW VRRP	172.18.1.197	
		IFW01	172.18.1.198	
		IFW02	172.18.1.199	
		INSIDEMX01-M0	172.18.1.200	
		INSIDEDNS01-M0	172.18.1.201	

1005	**10.0.0.0/23**	**corporate network connection**	**10.0.0.1-10.0.1.254**	
		IFW VRRP	10.0.0.1	
		IFW01	10.0.0.2	
		IFW02	10.0.0.3	
		INSIDEMX01-SIP	10.0.0.4	
		INSIDEDNS01-SIP	10.0.0.5	
		CORPDESK01	10.0.0.6	corporate desktop 1
		CORPDESK02	10.0.0.7	corporate desktop 2
		..		..
		CORPDESKXX	10.0.yy.xx	corporate desktop XX

OTHER		**OTHER IP's**		
	ISP secondary DNS	ISP_DNS	2.0.0.23	IANA reserved / picked
	Supplier A VPN endpoint	SupplierA_VPNGateway	5.3.4.2	IANA reserved / picked
	Supplier A database	SupplierA_DB	5.3.4.7	IANA reserved / picked
	Supplier B VPN endpoint	SupplierB_VPNGateway	7.2.3.5	IANA reserved / picked
	Supplier B database	SupplierB_DB	7.2.3.7	IANA reserved / picked
	Reseller A VPN endpoint	ResellerA_VPNGateway	23.4.1.5	IANA reserved / picked
	Reseller A database	ResellerA_DB	23.4.1.7	IANA reserved / picked
	Reseller B VPN endpoint	ResellerB_VPNGateway	41.5.3.2	IANA reserved / picked
	Reseller B database	ResellerB_DB	41.5.3.7	IANA reserved / picked

1.5. Required Skillsets

This section should list the basic requirements that should be met by the administrators to be able to handle the components of this network in an efficient manner.

- system administration skills

Every administrator should be able to maintain a UNIX system by him/herself. This involves everything from setting up a system from scratch, maintaining it, optimizing it, automating reocurring events to being able to completely rebuild a system from scratch, harden it and put it back into production. It would be desired that network administrators also have at least basic skills in this category. Tasks will include to be primary contact for all system related matters. Also, system administrators will be part of the incident response team. Sufficient security knowledge is required to be able to harden a system according to standards.

- network administration skills

Aside from device-specific skills (network administrators need to have excellent in-depth knowledge about the router, switches and terminalserver), network administrators need to have in-depth knowledge about networking, subnetting, network management. A certain understanding of the workings of the firewalls is also desired. Sufficient security knowledge is required to be able to maintain the security configurations on the router and switches (access-lists etc).

- security and incident handling skills

The security administrators are responsible for the firewalls, implementing and enforcing security policies, maintaining the firewall rulebases, maintaining the SecurID facility (and supporting end users for that matter) and performing security audits in reasonable intervals. Also, they are responsible for the IDS infrastructure, IDS maintenance, monitoring of IDS events, etc. They are the core of the incident response team that will be formed. Primary contact for all security related matters. In-depth system administration skills are required in order to maintain the security related servers (firewall management, SecurID servers, IDS sensor/analyzer, syslog server).

- application specific skills

In-depth application application specific skills are required for the employees that handle the web content, database servers, mail servers etc. System administrators are also application administrators to the extent that they have to handle the services. Staff would be also needed to maintain the content of the web presence, maintain the database server contents etc.

1.6. Access requirements/restrictions

1.6.1. Customers

(Companies or individuals that purchase bulk online fortunes)

Customers (coming from the Internet) only have access to the Web- and DNS-servers from the Internet. Everything else is not reachable, since it is blocked by the firewall. Customers really only need access to the Web- and DNS servers since all transactions etc are done per http/https.

1.6.2. Suppliers and Resellers

(Suppliers: companies that supply GIAC Enterprises with their fortune cookie sayings)
(Resellers: international companies that translate and resell fortunes)

The suppliers and resellers have access to the FTP servers in the external eKookie network.. the FTP server is ONLY accessible through the site-to-site VPNs GIAC will build with the resellers and suppliers. FTP is a high-risk protocol (clear text passwords etc) and needs to be secured however possible. It was suggested to use secure means like ssh and/or sftp instead, but it was determined that the resellers and suppliers may not be able to support that. Some would use applications that require ftp and would not be able to handle other protocols/services like ssh.

Suppliers will only be allowed to upload files into the FTP server, not download files from the FTP server. This will be enforced locally on the FTP server itself.

From an network standpoint, partners will be granted the same access as the suppliers: to the Web- and DNS-servers. The difference is, partners will only be allowed to download files from the FTP server, not upload anything to the FTP server.

Some sample access configuration can be seen in the rulebase (SupplierA and SupplierB site-to-site VPN endpoints).

1.6.3. GIAC Enterprises employees located on GIAC Enterprise's internal network

Access to the management networks is regulated by local access restrictions (accounts etc on the servers) and rules in the firewalls. It was desired that employees should be able to manage the servers from the corporate network, as well as through a VPN client.

1.6.4. GIAC Enterprises mobile sales force and teleworkers

The mobile sales force and teleworkers will use a VPN client to connect to the internal networks of the eKookie network. They will be allowed to connect to internal services, like corporate email, Intranet-web etc through the VPN. All VPN clients connect to the outside firewall cluster.

1.6.5. Employees Access to the outside world

Employees and other users that are coming from the corporate network and will be able to use the World Wide Web (HTTP), SSL-http (HTTPS) and messenger services like AOL Instant Messenger and/or ICQ. While the risk of using public chat systems like AIM and/or ICQ have been made aware of, it was decided that a messenger service is needed for workflow/communication flow efficiency reasons. It was suggested to GIAC to consider purchasing an internal messenger system that is restricted for internal use only. This is being considered. This proposal will not go into this, as requested by GIAC.

Administrators will also be able to ping out to the Internet for troubleshooting.

1.6.6. Additional Access Requirements

- all employees that need to use a VPN client and/or need access to production equipment will receive a SecurID fob to authenticate. Users without SecurID fob will not be able to log on to any production equipment
- access to workstations and non-production equipment will be controlled by MIS. Passwords will be changed in pre-set intervals, security audits on the workstations (open shares etc) will be undertaken in certain intervals as well. It was considered deploying SecurID on workstations as well, but cost and effort would have been too high (additional fobs required for EVERY single employee, maintenance of workstations etc).

-

2. Assignment 2 – Security Policies

Based on the security architecture that you defined in Assignment 1, provide a security policy for the following three components:

- Border Router(s)
- Primary Firewall(s)
- VPN(s)
- You may optionally include policy for other devices (i.e., - internal firewalls).

By "policy" we mean the specific set of ACLs, ruleset, or IPSec policy for that device – not corporate or organizational policy (though note that organizational policy may dictate the specific ACLs or ruleset in effect).

For each component, be sure to consider the access requirements for customers, suppliers, partners, remote users, and internal users that you defined in Assignment 1. The policies you define must accurately reflect those business needs as well as appropriate security considerations.

You must include the complete policy (meaning explicit ACLs, Ruleset, IPSec policy, etc.) in your paper. It is not enough to simply state "I would include ingress and egress filtering…" The policies may be included in an Appendix if doing so will help the "flow" of the paper (clearly state if this is the case).

For each rule in all policies, you must include the general purpose of the rule and why it is important.

You must also include a discussion of the order of the rules, and why order is (or is not) important.

For one of the three security policies defined above, you must create a clearly labeled & separate tutorial on how to implement the policy. This tutorial is in addition to the full policy for that device / function. Use screen shots network traffic traces, firewall log information, and/or URLs to find further information to clarify your instructions. Be certain to include a general explanation of the syntax or format of the ACL, filter, or rule for your device, as well as a general explanation of how to apply a given ACL, filter, or rule.

Be certain to point out any tips, tricks, or potential problems

2.1. Border Router

As mentioned above, GIAC will have an Internet connection with a bandwidth of 3MBit/sec. To achieve this, 2 T1's are linked together [MULITLINK] and referred to as a new interface, in our case called Multilink1. This interface refers to the two Serial interfaces that represent each T1 in our connection.

The router does basic filtering of known invalid addresses/networks, has certain unused services disabled and displays a banner message that states that only authorized employees are allowed to access that router.

The following lists the commands that are used in the router, together a short explanation of what each command means.

The inbound access list will be applied to the Multilink1 interface (traffic coming from the Internet to the GIAC network)

The outbound access list will be applied to the FastEthernet0/0 interface that is the network connection to the GIAC network (for traffic coming from the GIAC network going to the Internet).

[CISCO1], [CISCO2], [CISCO3]

2.1.1. Configuration

2.1.1.1. general configuration on the border router

```
logging 172.18.1.5
```
- log syslog messages to the syslog server (via external management network)

```
service password-encryption
```
- encrypt displayed passwords

```
no ip bootp server
```
- disable bootp server

```
no ip http server
```
- disable http server

```
no ntp master
```
- disable ntp service on router

```
no ip source-route
```
- disable source routing to fight IP spoofing

```
no service tcp-small-servers
no service udp-small-servers
```
- Ports below 20 TCP and UDP and time TCP and UDP port 37. Rarely needed, should be disabled

```
banner motd # You have logged on to a GIAC proprietary device.
INFORMATION IN THIS DEVICE BELONGS TO GIAC AND/OR ONE OF ITS
```

```
AUTHORIZED CLIENTS AND MAY NOT BE COPIED (IN WHOLE OR IN PART) IN
ANY MANNER WITHOUT EXPRESS WRITTEN AUTHORIZATION. This device may
be used only for the authorized business purposes of GIAC and/or
its clients. Anyone found using this device or its information
for any unauthorized purpose or personal use may be subject to
disciplinary action and/or prosecution.
#
```

- set the banner. (Cisco Router Hardening Step by Step, Dana Graesser [BANNER]).

```
no service finger
```

- finger service, disable. Finger is a service that allows users to query a server to find out information about a user by their email address. Depending on the finger server, you may find out only if the user is currently logged on, or personal information including the last time they retrieved their mail, telephone number, full name, address, etc. Should be disabled for privacy reasons.

```
no cdp run
```

- disable CDP, Cisco Discovery Protocol. Not needed.

```
ntp disable
```

- disable incoming ntp

```
no snmp
```

- disable incoming snmp queries from the Internet

```
no ip direct-broadcast
```

- This command disallows echo requests to broadcast addresses, preventing amplification of "smurf" type attacks, where one packet can generate hundreds of responses.

```
no ip redirects
```

- This command is often used in conjunction with anti-spoofing access-lists and is applied to any router interface from which malicious traffic may enter your network

```
no ip unreachables
```

- Don't send icmp for denied items in access-list.

2.1.1.2. Inbound access lists

```
ip access-list extended filterin
```

- begin of access list "filterin" (inbound from the Internet)

```
deny ip 10.0.0.0 0.255.255.255 any log
deny ip 172.16.0.0 0.15.255.255 any log
deny ip 192.168.0.0 0.0.255.255 any log
```

- deny RFC1918 IPs from entering the network from the Internet; most likely spoofed IPs

```
deny ip 127.0.0.0 0.255.255.255 any log
```

- deny "localhost" from entering the network; this is not possible, traffic must be spoofed

```
deny ip host 0.0.0.0 any log
```

- deny packets without ip address.

```
permit udp any host 1.1.1.148 53
```

- allow Internet -> primary DNS for DNS queries

```
permit tcp any host 1.1.1.149 80
permit tcp any host 1.1.1.149 443
```
- allow http and https from the Internet to the web server

```
permit tcp any host 1.1.1.151 25
```
- allow mail traffic from the Internet to the SMTP relay

```
permit tcp host 128.2.136.71 host 1.1.1.148 123
permit udp host 128.2.136.71 host 1.1.1.148 123
```
- allow ntp traffic (response) from ntp-1.ece.cmu.edu towards the DNS server

```
permit tcp host 128.59.59.177 host 1.1.1.148 123
permit udp host 128.59.59.177 host 1.1.1.148 123
```
- allow ntp traffic (response) from sundial.columbia.edu towards the DNS server

```
permit tcp host 216.27.190.202 host 1.1.1.148 123
permit udp host 216.27.190.202 host 1.1.1.148 123
```
- allow ntp traffic (response) from time.berkeley.netdot.net towards the DNS server

```
permit tcp any host 1.1.1.145 264
```
- allow SecuRemote traffic (264=FW1_topo for topology downloads)

```
permit udp any host 1.1.1.145 500
```
- allow IKE traffic (SecuRemote, VPN tunnels)

```
permit esp any host 1.1.1.145
```
- allow ESP traffic (SecuRemote, VPN tunnels)

```
permit tcp host 2.0.0.23 host 1.1.1.148 53
```
- allow zone transfers from our primary DNS to the ISP secondary DNS

```
permit icmp any any packet-too-big
```
- allow packet too big messages for getting the MTU size right

```
permit tcp any 80 host 1.1.1.152
permit tcp any 443 host 1.1.1.152
permit tcp any 5190 host 1.1.1.152
permit icmp any echo-reply host 1.1.1.152
```
- allow response traffic from the Internet to the corporate network for http,https,AIM, echo replies traffic (hidden behind NAT address)

```
deny ip any any log
```
- deny everything else and log

2.1.1.3. Outbound access lists

```
ip access-list extended filterout
```
- begin of access list "filterout" (outbound to the Internet)

```
deny ip 10.0.0.0 0.255.255.255 any log
```

```
deny ip 172.16.0.0 0.15.255.255 any log
deny ip 192.168.0.0 0.0.255.255 any log
```
- deny RFC1918 IPs from entering the network from the Internet; most likely spoofed IPs

```
deny ip 127.0.0.0 0.255.255.255 any log
```
- deny "localhost" from entering the network; this is not possible, traffic must be spoofed

```
deny ip host 0.0.0.0 any log
```
- deny packets without ip address

```
deny icmp any any time-exceeded
```
- deny icmp time-exceeded messages from leaving the network, frequently used by traceroute to map networks. Don't log, can be many messages and not too useful to log anyway

```
permit tcp host 1.1.1.148 host 2.0.0.23 53
```
- allow responses to zone transfers for the ISP secondary DNS from our primary DNS

```
permit udp host 1.1.1.148 53 any
```
- allow responses to Internet -> primary DNS for DNS queries

```
permit udp host 1.1.1.148 any 53
```
- allow DNS lookups from primary DNS to Internet DNS servers

```
permit tcp host 1.1.1.149 80 any
permit tcp host 1.1.1.149 443 any
```
- allow responses for http and http requests from the Internet to the Webserver

```
permit tcp host 1.1.1.151 25 any
```
- allow responses to SMTP traffic from the Internet to the MX

```
permit tcp host 1.1.1.151 any 25
```
- allow sending of mail to Internet mail servers

```
permit tcp host 1.1.1.148 host 128.2.136.71 123
permit udp host 1.1.1.148 host 128.2.136.71 123
```
- allow ntp traffic from the primary DNS to ntp-1.ece.cmu.edu

```
permit tcp host 1.1.1.148 host 128.59.59.177 123
permit udp host 1.1.1.148 host 128.59.59.177 123
```
- allow ntp traffic from the primary DNS to sundial.columbia.edu

```
permit tcp host 1.1.1.148 host 216.27.190.202 123
permit udp host 1.1.1.148 host 216.27.190.202 123
```
- allow ntp traffic from the primary DNS to time.berkeley.netdot.net

```
permit tcp host 1.1.1.145 264 any
```
- allow firewall cluster to respond to FW1_topo requests from the Internet

```
permit udp host 1.1.1.145 500 any
```
- allow the firewall cluster to respond to IKE/VPN requests from the Internet, as well as to send IKE packets to initiate tunnels

```
permit esp host 1.1.1.145 any
```
- allow the firewalls to respond to ESP packets for VPN

```
permit tcp host 1.1.1.152 any 80
permit tcp host 1.1.1.152 any 443
permit tcp host 1.1.1.152 any 5190
permit icmp host 1.1.1.152 any echo-request
```
- allow http,https, AIM,ping traffic from the internal corporate network (hidden behind NAT address) to the Internet

```
permit icmp any any packet-too-big
```
- allow packet too big messages for getting the MTU size right

```
deny ip any any log
```
- deny everything else and log

```
interface Multilink1
[…]
      ip access-group filterin in
 […]
```

- apply access lists to Multilink1 interface

```
interface FastEthernet0/0
[…]
      ip access-group filterout out
[…]
```

- apply access lists to FastEthernet interface

2.1.1.4. SecurID support on the Border Router

```
aaa new-model
```
This command enables the AAA access control system

```
aaa authentication login default enable
```
The default and optional list names that you create with the aaa authentication login command are used with the login authentication command

To create a default list that is used if no list is assigned to a line, use the login authentication command with the default argument followed by the methods you want to use in default situations
-> `enable` - Uses the enable password for authentication

```
aaa authentication login securid1 group tacacs+ line
```
define an authentication list called "securid1" and use tacacs+ as primary authentication method, then the line password if tacacs+ does not work

```
aaa authentication enable default group tacacs+ enable
```
to go into privileged/enable mode, use tacacs+ as primary and the set enable password as backup authentication method, if tacacs+ does not work

```
tacacs-server host 172.18.1.21
tacacs-server host 172.18.1.22
```
list servers that support TACACS+ (NOTE: RSA ACE/server (SecurID01 and 02) supports TACACS+ and RADIUS (RADIUS will be used as authentication method for accessing the Terminalserver / server console ports)

```
tacacs-server timeout 30
```
30 second timeout. After 30 seconds, TACACS+ will be considered failed and the next chosen authentication method will be used

```
tacacs-server key s3cr3t
```
pre-shared key for TACACS+ server-router communication

```
line aux 0
access-class 2 in
transport input all
access-list 2 deny 0.0.0.0 255.255.255.255
```
- Block access to aux

```
line vty 0 4
access-class 1 in
transport line input ssh
login authentication securid1
```
- allow ssh connections, use authentication list "securid1" for list of methods available

```
password 7 xxxxxxxxxxxxx
login
```
- require password authentication

- allow only specific networks to ssh into router:

```
access-list 1 permit 172.18.0.64 0.0.0.224
```
- vpn pool network; Network Admins coming in with the VPN client

```
access-list 1 deny ip any any log
```
- drop everything else and log

2.1.1.5. syslogging and SecurID authentication traffic

Packets that hit any deny rule will get logged (everything but time-exceeded messages which are traceroute attempts, happen quite often and generate a lot of noise). The router will log to the SYSLOG01 service via the SYSLOG01's service IP. In order to get there, it will send the traffic towards the IP of the VRRP of the outside firewall cluster in the external device management VLAN and will get handled from there. That remedies the potential problem of the router not being able to get to internal IPs in the GIAC network.

This also applies to the TACACSplus SecurID traffic: the traffic will go across the external management VLAN towards the outside firewall cluster and will get forwarded to the inside networks.

2.1.2. Rule Order

You need to pay attention to grouping the rules in the way they should be applied (are they valid globally, only on one interface, or on a line like the ssh session?). Also, access lists are organized top-bottom, meaning that the first match counts and will be applied against the packet/traffic. For example:

```
access-list 1 permit 172.18.0.64 0.0.0.224
```
- vpn pool network; Network Admins coming in with the VPN client

```
access-list 1 permit 10.0.0.0 0.0.1.255
```
- for mgmt connections coming from the corporate network

```
access-list 1 deny ip any any log
```
- drop everything else and log

If in this case the deny ip any any log would be on top of the other 2 access lists, no traffic would be allowed, everything would get dropped. It is important to know this fact!

2.2. VPN / core firewall policies

2.2.1. Introduction

This section contains a complete list of rules that will be implemented on the outside and inside firewall clusters. Every rule has a comment field that explains the reason for that specific rule.

The policy is organized into different sections:

- firewall specific rules (FW management, site-to-site VPNs, basic VPN client access rules (e.g. for topology download)
- VPN client specific rules (which group of users has which basic access)
- Server specific / network general rules (server – server communication, network – server / network communication)

2.2.2. VPN details

The only IPSEC/encryption scheme that is supported by Firewall-1 NG FP3 is IKE. AH is also no longer supported by NG, ESP as security protocol identifier (SPI) is selected as default. The Security Association type that needs to be used here is the tunnel mode. Reason: at least our side of the VPN endpoint / security association is a gateway.

2.2.3. Split Tunneling / VPN client side

There are two types of VPN clients that can be used with Checkpoint Firewall-1: SecuRemote and SecureClient. SecureClient supports local desktop policies that can be restricted so that a user connected via the VPN client cannot go to any other non-internal destination. The policy is pulled from a policy server, oftentimes the policy server and the VPN gateway are on the same machine. While this is desirable, separate licensing is needed for the SecureClient setup (client and policy server license), and the cost associated with this is significant.

SecuRemote on the other hand does not need separate licensing; a 100-user license comes with the purchase of the 3DES/STRONG encryption license for the VPN gateway. Disadvantage: it supports split tunneling. While this is actually an advantage for the user (he/she can surf the Internet while connected to the internal network, i.e. check email and research in the Web at the same time), this poses a tremendous risk for the security of the internal network(s): the security perimeter is expanded to the user's desktop when he/she is connected via a VPN client. Should the user machine get compromised for any reason, the intruder may have direct access into the internal network(s), depending on the user's permissions.

To address these issues, the following compromise was done: a host-based firewall is installed on the user's laptop by default, default user permissions on his/her laptop are set so he/she cannot tamper with the configuration of the machine, cannot install/uninstall programs etc etc. Access permissions may only be modified by manager's approval, if there is a business reason for it. The laptop configuration is managed by the MIS department of the company and regulated by corporate security policy.

2.2.4. Explanation of the Naming Conventions used in the Policies

(* stands for placeholder)

2.2.4.1. Groups

*_ALL characterizes a group of objects
*_Mgmt_ALL contains the management interface IPs of objects
*_SIP_ALL contains service interface IPs of objects

2.2.4.2. Nodes

NODE-E0 stands for publicly reachable IP address of node
NODE-Ex stands for other public IP address of node
NODE-M0 stands for mgmt IP address of node
NODE-SIP stands for multipath service IP address of node
NODE-VRRP stands for VRRP address of NODE-cluster
NODE-NAT stands for NATed address of internal node

2.2.5. Outside Firewall Policy

2.2.5.1. Rule 1

Clusters using VRRP (like the firewalls) need to send VRRP multicast messages every so often. Since this is benign traffic and happens a lot, it does not need to get logged.

2.2.5.2. Rule 2

FW1 for state syncing between the active and the passive firewall, ssh for being able to reach each firewall in case the interface facing the management etc would be down for any reason.

2.2.5.3. Rule 3

CPD_amon for application/fw monitoring, FW1 for control/ pushing policy, FW1_ica_services for NG certificate validation, ssh for access.

2.2.5.4. Rule 4

FW1_log for logging to the management server that also acts as a log server at the same time, FW1 for pulling the policy upon restart.

2.2.5.5. Rule 5

This is for SecuRemote VPN client connections. IKE/ESP for VPN tunnel, FW1_topo for FW-1 topology downloads.

2.2.5.6. Rules 6 through 10

6	OFWCluster ReselllerA_VPNGateway	OFWCluster ReselllerA_VPNGateway	IPSEC	accept	Log	OFWCluster	* Any	IKE/ESP/AH for site-to-site VPN
7	OFWCluster ReselllerB_VPNGateway	OFWCluster ReselllerB_VPNGateway	IPSEC	accept	Log	OFWCluster	* Any	IKE/ESP/AH for site-to-site VPN
8	OFWCluster SupplierA_VPNGateway	OFWCluster SupplierA_VPNGateway	IPSEC	accept	Log	OFWCluster	* Any	IKE/ESP/AH for site-to-site VPN
9	OFWCluster SupplierB_VPNGateway	OFWCluster SupplierB_VPNGateway	IPSEC	accept	Log	OFWCluster	* Any	IKE/ESP/AH for site-to-site VPN
10	Reseller_Databases_ALL Supplier_Databases_ALL	FTP01-SIP-E0	TCP ftp	Encrypt	Log	OFWCluster	* Any	- Resellers: download from FTP site - Suppliers: upload to FTP site all via VPN tunnels, encrypted.

These are the site-to-site VPN rules for the Supplier/Reseller <-> GIAC network to allow encrypted packets between those sites and the GIAC network. Note that in order to actually build the tunnel, shared secrets must be exchanged prior to trying to use the tunnel. Rule 10 then uses the actual tunnel, allowing FTP across to the FTP server, encrypted.

2.2.5.7. Rules 11 through 17

	VPN Client specific rules follow							
11	SystemAdministrators@Any	Public_Servers_Mgmt_ALL Inside_ProdServers_Mgmt_ALL	TCP ssh ping	Client Encrypt	Log	OFWCluster	* Any	Allow SysAdmins to ssh to the outside servers
12	NetworkAdministrators@Any	Outside_Device_Mgmt_ALL Inside_Device_Mgmt_ALL	TCP ssh ping	Client Encrypt	Log	OFWCluster	* Any	allow NetworkAdmins to ssh to outside switch cluster
13	SecurityAdministrators@Any	Outside_Mgmt_Networks Inside_ProdServers_Mgmt_ALL Security_Servers_Mgmt_ALL OFWCluster IFWCluster	ping TCP ssh	Client Encrypt	Log	OFWCluster	* Any	Security Admins need to have unrestricted access
14	SecurityAdministrators@Any	FWMGMT01	TCP CPMI TCP FW1_mgmt	Client Encrypt	Log	OFWCluster	* Any	firewall management through Client VPN
15	AllAdmins@Any	TSERVER01-M0	TCP TerminalServerPorts	Client Encrypt	Log	OFWCluster	* Any	allow Terminalserver access through VPN
16	TeleWorkers@Any	INSIDEDNS01-SIP	UDP domain-udp	Client Encrypt	Log	OFWCluster	* Any	TeleWorkers, mobile Salesforce - DNS access
17	TeleWorkers@Any	INSIDEMX01-SIP	TCP smtp	Client Encrypt	Log	OFWCluster	* Any	TeleWorkers, mobile Salesforce - Corporate Mail access

These are the rules that deal with SecuRemote VPN client connections, allowing access depending on group ownership / permissions.

2.2.5.8. Rule 18

	Server specific rules follow							
18	Outside_Mgmt_Networks	OFWCluster	ping	accept	Log	OFWCluster	* Any	ping firewall cluster from servers in outside network for testing purposes

This allows the servers and devices in the outside management networks to ping the outside firewall cluster, for network/interface testing and troubleshooting purposes.

2.2.5.9. Rule 19

19	* Any	OFWCluster OFW_ALL	* Any	drop	Log	OFWCluster	* Any	- stealth rule.. drop everything else to firewalls that was not allowed above

This is the stealth rule, dropping all other traffic destined for the outside firewalls and the cluster IP itself and log.

2.2.5.10. Rules 20 and 21

20	GIAC_networks_ALL	DNS01-SIP-E0	UDP domain-udp	accept	Log	OFWCluster	* Any	- DNS queries against our DNS
21	GIAC_networks_ALL	WWW01-SIP-E0	TCP http	accept	Log	OFWCluster	* Any	- public access to webserver

These are the public access rules for the DNS and WWW servers. These enables customers / machines from the Internet to query our publicly accessible services.

2.2.5.11. Rule 22

22	DNS01-SIP-E0	* Any	UDP domain-udp	accept	Log	OFWCluster	* Any	- DNS queries coming from our DNS server

This deals with the public/primary DNS, allowing access to anything in the Internet in order to resolve Internet-hosts, do reverse lookups for those hosts etc.

2.2.5.12. Rule 23

23	DNS01-SIP-E0	INET_NTP_Servers_ALL	ntp	accept	Log	OFWCluster	* Any	- NTP sync from Internet NTP

This allows the primary DNS to ntp sync to pre-determined NTP servers in the Internet, so internal servers can then sync up with our primary DNS.

2.2.5.13. Rule 24

24	Public_Servers_SIP_ALL	DNS01-SIP-E0	ntp UDP domain-udp	accept	Log	OFWCluster	* Any	- public servers querying NTP and domain-udp

This rule deals with the public servers to DNS communication for NTP and DNS queries.

2.2.5.14. Rule 25

25	INSIDEDNS01-SIP	DNS01-SIP-E0	TCP domain-tcp ntp	accept	Log	OFWCluster	* Any	- zone transfers from our external DNS to the internal DNS - ntp sync between internal DNS and public DNS

This rule allows the internal DNS to transfer zones from and ntp sync with the primary DNS.

2.2.5.15. Rule 26

26	ISP_DNS-E0	DNS01-SIP-E0	TCP domain-tcp	accept	Log	OFWCluster	* Any	- allow ISP DNS server to pull zone files from our primary public DNS

This allows the secondary DNS at our ISP to transfer the public zone from our primary DNS.

2.2.5.16. Rule 27

This is a backdoor for the inside DNS server. If the primary DNS is down and the public zone on the internal DNS would expire, as well as there would be the need to look up Internet hosts (e.g. corporate network users trying to access the Internet), the inside DNS is allowed to query the ISP DNS instead.

2.2.5.17. Rule 28

This rule allows mail exchange between the Internet and our public/primary MX relay.

2.2.5.18. Rule 29

This rule allows mail communication between the public and internal MX relays.

2.2.5.19. Rule 30

This rule deals with syslog traffic: allow syslogging to happen from the outside devices and public servers to the internal syslog server.

2.2.5.20. Rule 31

This rule allows SecurID authentication traffic to be sent to the internal SecurID servers, from the outside management networks and the firewalls (the latter need it to be able to authenticate VPN users per SecurID).

2.2.5.21. Rule 32

Cisco equipment does not support native SecurID. In place of that, authentication occurs via TACACSPLUS. This rule allows that communication.

2.2.5.22. Rule 33

Backup rule. This allows the communication between the Backup server and the public servers for Veritas Netbackup. Veritas Netbackup needs a range of ports to be open, in this case, 13720 through 13750/tcp.

2.2.5.23. Rule 34

web server-to-DB server communication via sqlnet. This is needed for the application running on the web server to retrieve customer records and fortune cookie information from the Oracle DB.

2.2.5.24. Rule 35

DB – to – FTP server communication per ssh. Needed for copying fortune cookie saying to/from the FTP server.

2.2.5.25. Rule 36

Corporate network access to manage/access all public servers per ssh.

2.2.5.26. Rule 37

Corporate network access to the Internet, for http, https, AIM/AOL. The corporate network is hidden behind the "CORP-NAT" (discussion about NATting later on in this assignment).

2.2.5.27. Rule 38

Cleanup rule. Drop everything else that has not been explicitly allowed and log.

2.2.6. Inside Firewall Policy

2.2.6.1. Rule 1

	IFW specific rules follow							
1	* Any	multicastnets	vrrp igrp	accept	- None	IFWCluster	* Any	- vrrp multicasting (for firewall and switch cluster VRRP communication)

This allows VRRP multicast messages to be sent from the inside devices like the firewalls. Lot of benign messages, will not get logged.

2.2.6.2. Rule 2

2	IFW_ALL	IFW_ALL	ping TCP ssh TCP FW1	accept	Log	IFWCluster	* Any	- FW management connection for state sync, - ssh/ping as well

FW1 for state syncing, ssh for emergency access.

2.2.6.3. Rule 3

3	FWMGMT01	IFW_ALL OFW_ALL	TCP CPD_amon TCP FW1 TCP ssh TCP FW1_ica_services	accept	Log	IFWCluster	* Any	- management of firewalls - FW1 for pushing policies - ssh for access

FW-1 management server communication to the firewalls, for monitoring (CPD_amon), pushing policies and control messages (FW1), validating NG certificates (FW1_ica_services) messages, ssh for access.

2.2.6.4. Rule 4

4	IFW_ALL OFW_ALL	FWMGMT01	TCP FW1_log TCP FW1	accept	Log	IFWCluster	* Any	- fw traffic logging to management server - FW1 protocol for fetching policy upon startup

Firewall to management server communication for logging (FW1_log) and pulling policies (FW1).

2.2.6.5. Rules 5 through 11

5	OFW_Pool_Network	Inside_ProdServers_Mgmt_ALL	TCP ssh ping	accept	Log	IFWCluster	* Any	Allow SysAdmins to ssh to the outside servers
6	OFW_Pool_Network	Inside_Device_Mgmt_ALL	TCP ssh ping	accept	Log	IFWCluster	* Any	allow NetworkAdmins to ssh to the outside switch cluster
7	OFW_Pool_Network	Security_Servers_Mgmt_ALL IFWCluster	ping TCP ssh	accept	Log	IFWCluster	* Any	Security Admins need to have unrestricted access
8	OFW_Pool_Network	FWMGMT01	TCP CPMI TCP FW1_mgmt	accept	Log	IFWCluster	* Any	firewall management through Client VPN
9	OFW_Pool_Network	TSERVER01-M0	TCP TerminalServerPorts	accept	Log	IFWCluster	* Any	allow Terminalserver access through VPN
10	OFW_Pool_Network	INSIDEDNS01-SIP	UDP domain-udp	accept	Log	IFWCluster	* Any	TeleWorkers, mobile Salesforce - DNS access
11	OFW_Pool_Network	INSIDEMX01-SIP	TCP smtp	accept	Log	IFWCluster	* Any	TeleWorkers, mobile Salesforce - Corporate Mail access

These rules deal with SecuRemote VPN connections. They allow needed access to the internal networks for VPN users. We cannot list user groups here. Reason is that the inside firewall cluster will not receive

this traffic encrypted, it will be decrypted at the outside firewall cluster (since the VPN tunnel for the client terminates there, not at the inside firewall cluster). Once decrypted, it gets assigned an IP-NAT from an IP pool range dedicated for VPN connections. Hence, the inside firewall cluster only sees IPs from that VPN IP pool, rather than the user names. That is what we have to filter on. This means however, that users getting assigned an IP pool NAT otherwise (system adminstrators etc) are also able to ssh to devices that should only be accessible by security staff (e.g. security servers, as can be seen in the related rules [13 etc] in the outside firewall policy). The filtering then has to occur on a server-level, with shell accounts (only the staff that actually has shell accounts / the permissions to ssh into certain servers can do so. That is another reason why securing a network has to happen on multiple layers.

2.2.6.6. Rule 12

This is for servers and devices in the inside networks to be able to ping the firewalls in order to test the network and/or troubleshoot issues.

2.2.6.7. Rule 13

Stealth rule. Drop everything else that is destined for the inside firewall cluster and the VRRP of the firewall cluster and log.

2.2.6.8. Rule 14

This rule allows the internal DNS to transfer zones from and ntp sync with the primary DNS.

2.2.6.9. Rule 15

This rule deals with the inside/internal servers to DNS communication for NTP and DNS queries.

2.2.6.10. Rule 16

Backdoor for the inside DNS server. If the primary DNS is down and the public zone on the internal DNS would expire, as well as there would be the need to look up Internet hosts (e.g. corporate network users trying to access the Internet), the inside DNS is allowed to query the ISP DNS instead.

2.2.6.11. Rule 17

17	MX01-SIP-E0 INSIDEMX01-SIP	INSIDEMX01-SIP MX01-SIP-E0	TCP smtp	accept	Log	IFWCluster	* Any	- allow smtp communication between mail relay and internal mail server

This rule allows mail communication between the public and internal MX relays.

2.2.6.12. Rule 18

18	Inside_Device_Mgmt_ALL Inside_ProdServers_Mgmt_ALL Outside_Device_Mgmt_ALL Public_Servers_Mgmt_ALL	SYSLOG01-SIP	UDP syslog	accept	Log	IFWCluster	* Any	- Servers/Devices log to syslog

Syslog traffic: allow syslogging to happen from all devices and public servers to the internal syslog server.

2.2.6.13. Rule 19

19	Outside_Mgmt_Networks Inside_Mgmt_Networks OFW_ALL	SECURID_SIP_ALL	UDP securid-udp	accept	Log	IFWCluster	* Any	- server to SecurID servers for SecurID authentication traffic - OFW cluster to SecurID for VPN client-SecurID authentication

SecurID authentication traffic to the internal SecurID servers, from all management networks and the outside firewalls (the latter need it to be able to authenticate VPN users per SecurID).

2.2.6.14. Rule 20

20	Outside_Device_Mgmt_ALL Inside_Device_Mgmt_ALL	SECURID_SIP_ALL	TCP TACACSplus	accept	Log	IFWCluster	* Any	- router and switches talk TACACS+ to the SecurID servers for authentication

Cisco equipment does not support native SecurID. In place of that, authentication occurs via TACACSPLUS. This rule allows that communication.

2.2.6.15. Rule 21

21	TSERVER01-M0	SECURID_SIP_ALL	UDP RADIUS	accept	Log	IFWCluster	* Any	- Terminalserver talks RADIUS to the SecurID servers for authentication

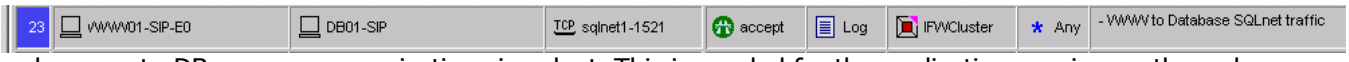
The terminalserver does not support native SecurID or TACACSplu, only RADIUS. This rule allows this traffic.

2.2.6.16. Rule 22

22	TAPE01-SIP Inside_ProdServers_Mgmt_ALL Public_Servers_Mgmt_ALL	Public_Servers_Mgmt_ALL Inside_ProdServers_Mgmt_ALL TAPE01-SIP	TCP NetBackup-Ports	accept	Log	IFWCluster	* Any	- initiate backup of servers

Communication between the Backup server and the public servers for Veritas Netbackup. Veritas Netbackup needs a range of ports to be open, in this case, 13720 through 13750/tcp.

2.2.6.17. Rule 23



web server-to-DB server communication via sqlnet. This is needed for the application running on the web server to retrieve customer records and fortune cookie information from the Oracle DB.

2.2.6.18. Rule 24

DB – to – FTP server communication per ssh. Needed for copying fortune cookie saying to/from the FTP server.

2.2.6.19. Rule 25

Corporate network access to manage/access all servers per ssh.

2.2.6.20. Rule 26

Corporate network access to the Internet, for http, https, AIM/AOL.

2.2.6.21. Rule 27

Cleanup rule. Drop everything else that has not been explicitly allowed and log.

2.2.7. NATting

Below you can see the network address translation that takes place in the GIAC network.

Rules 1 and 6 are manually created, rules 2 through 5 are automatically generated by assigning static NATs to the objects themselves. FW-1 always translates the primary/original IP into the static NAT unless it gets told to handle the object differently under certain circumstances. All those rules are installed (in the policy) on the OFWCluster and IFWCluster at the same time. In this case, rule number 1 means: do not use static NATs in communication that happens within the GIAC network itself.

Corporate network IPs are hidden behind the CORPNAT IP address, a hide-mode NAT (many-to-one NAT). The translation occurs on the outside firewall cluster. Internally, corporate network machines talk to other machines within the GIAC network by using their original 10-net address. Every server and device has appropriate routing setup, so that traffic coming from the 10-network gets sent back to the appropriate firewall cluster. Outside devices/servers (GIAC webserver, outside switch etc) send the traffic to the outside firewall cluster, which in turn routes it to the internal firewall cluster. Inside devices/servers (FW mgmt etc) route it to the internal firewall cluster directly. In any way, the IFWcluster then sends the traffic directly to the 10-net-machine.

Employees and other users who come from the inside of the network and desire to access public resources and the Internet come through the internal firewall cluster get translated into the public IP address of the internal firewall cluster, routed through the outside firewall cluster. The response hits the outside cluster, gets translated back into the real source address, gets routed to the inside cluster, and gets forwarded to the client that initiated the communication.

The GIAC_networks_ALL contains the encryption domain of the outside firewall cluster; all the networks that are behind the OFWCluster and for which the OFWCluster should handle the traffic. In our case:

- the outside firewalls themselves
- the corporate 10 network
- the RFC1918 networks (172.18.0.0 and 172.18.1.0)
- the public server network 1.1.1.144/28. Note, it does not contain the networks in front of the firewall cluster (1.1.1.0/28 and 1.1.1.128/28), simply because the OFWCluster does not have control over these networks

2.2.7.1. Proxy ARP

In order to be able to respond to the traffic destined for the public NAT, a so called proxy ARP needs to be installed/setup on the OFWCluster. On Nokia appliances, this is done per Voyager, the Nokia configuration tool for the firewall appliance. The virtual MAC address of the public VRRP address needs to be linked with the NAT address, so that ARP requests for that IP address get to the firewall.

Security - giac_ifw_021201_01 | Address Translation - giac_ifw_021201_01

NO.	ORIGINAL PACKET			TRANSLATED PACKET			INSTALL ON	COMMENT
	SOURCE	DESTINATION	SERVICE	SOURCE	DESTINATION	SERVICE		
1	GIAC_networks_ALL	GIAC_networks_ALL	* Any	= Original	= Original	= Original	IFWCluster OFWCluster	dont translate for servers coming from the inside going to servers on our networks
2	INSIDEDNS01-SIP	* Any	* Any	INSIDEDNS01-SIP	= Original	= Original	IFWCluster	Automatic rule (see the network obje data).
3	* Any	INSIDEDNS01-SIP (Valid	* Any	= Original	INSIDEDNS01-SIF	= Original	IFWCluster	Automatic rule (see the network obje data).
4	INSIDEMX01-SIP	* Any	* Any	INSIDEMX01-SIP (\	= Original	= Original	IFWCluster	Automatic rule (see the network obje data).
5	* Any	INSIDEMX01-SIP (Valid /	* Any	= Original	INSIDEMX01-SIP	= Original	IFWCluster	Automatic rule (see the network obje data).
6	Corporate-10-Network	* Any	* Any	CORP-NAT	= Original	= Original	OFWCluster	corporate network to internet: transl into hide mode NAT

Dynamic ARP Entries:
There are currently 162 dynamic ARP entries.

Display or Remove Dynamic ARP Entries

Click to flush all dynamic ARP entries: Flush

Proxy ARP Entries

Delete	IP Address	Interface	MAC Address
☐	1.1.1.152	User-defined MAC address	0:0:5e:0:1:10
☐	1.1.1.153	User-defined MAC address	0:0:5e:0:1:10
☐	1.1.1.154	User-defined MAC address	0:0:5e:0:1:10

Sample Voyager configuration

Above you can see a snapshot on how we would configure the outside firewalls OFW01 and OFW02 to reflect the public NATs by configuring the proxy arp entries. The MAC address has to be the virtual MAC address of the VRRP IP in the public VLAN. The virtual MAC address can be seen by executing an `ifconfig -a` on the active firewall in the VRRP pair:

```
IPSO 3.5-FCS10 #1041: 01.26.2002 202900
OFW01# ifconfig -a
eth-s1p1c0:  lname eth-s1p1c0
flags=e7<UP,PHYS_AVAIL,LINK_AVAIL,BROADCAST,MULTICAST,AUTOLINK>
        inet mtu 1500
        inet 1.1.1.145/28 broadcast 1.1.1.159 vrrpmac 0:0:5e:0:1:10
        inet 1.1.1.146/28 broadcast 1.1.1.159
        phys eth-s1p1 flags=4133<UP,LINK,BROADCAST,MULTICAST,PRESENT>
        ether 0:a0:8e:40:eb:78 speed 100M full duplex
```

2.2.8. Rule order

As with the access-lists in the border router, we pay attention to sort the rules in a way that most optimally reflects the frequency in which the rules are hit; the rules that get the most traffic should be before rules that get hit less. Especially in high-traffic situations, this way of organizing the rules will prove to increase the performance of the firewalls.

2.3. Tutorial

2.3.1. Rule Syntax

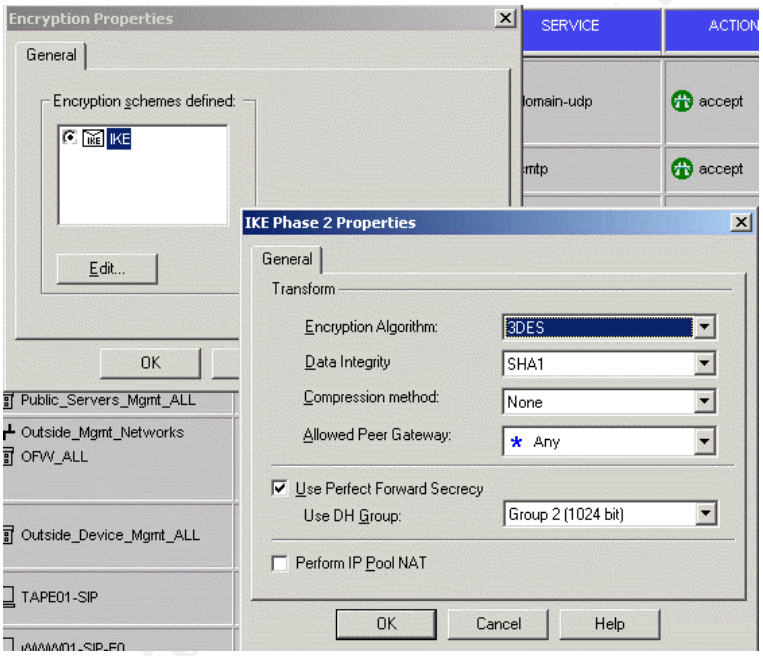
NO.	SOURCE	DESTINATION	SERVICE	ACTION	TRACK	INSTALL ON	TIME	COMMENT
28	MX01-SIP-E0	INSIDEMX01-SIP	TCP smtp	accept	Log	OFWCluster	* Any	- then from the relay to the inside mail server
29	INSIDEMX01-SIP	* Any	TCP smtp	accept	Log	OFWCluster	* Any	- inside mail server can connect to mail servers in the Internet
30	Reseller_Databases_ALL Supplier_Databases_ALL	FTP01-SIP-E0	TCP ftp	Encrypt	Log	OFWCluster	* Any	- Resellers: download from FTP site - Suppliers: upload to FTP site all via VPN tunnels, encrypted.
31	Outside_Device_Mgmt_ALL Public_Servers_Mgmt_ALL	SYSLOG01-SIP	UDP syslog	accept	Log	OFWCluster	* Any	- Servers/Devices log to syslog

Rules in the the Checkpoint SmartDashboard / Policy Editor are displayed as can be seen above:

rule no. | source | dest. | service | action | track | install target | time rule is valid | comment

1) rule number. The rule number is assigned by the gui-client.
2) source. Sources can be:
    a. objects/hosts/networks/groups
    b. usergroups
3) destination. Destinations can be objects/hosts/networks/groups
4) service. any sort of TCP/UDP/userdefined services and service-group

5) action to take. Can be:
    a. accept
    b. drop
    c. reject
    d. user/client/session authentication
    e. encrypt (for site-to-site vpn connections)
        - it is recommended to have the following settings for VPN tunnels: ESP, 3DES, SHA1, Perfect Forward Secrecy for maximum security. Peer Gateway should be set to the other end of the site-to-site VPN. In the screenshot, multiple VPN endpoints were used, so it had to be set to ANY. In NG, AH is not supported any more; the encapsulation method is set to ESP by default



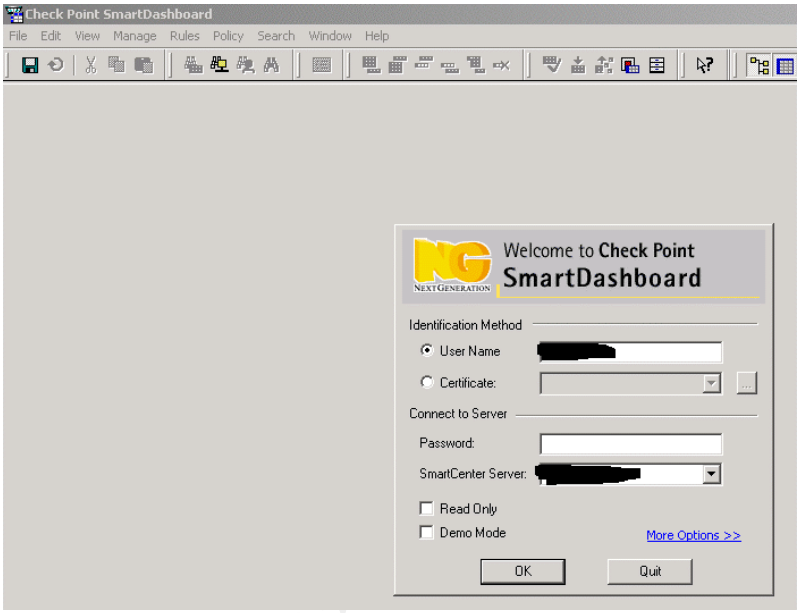
    f. client encrypt (for SecuRemote VPN client connections)

6) Track options are:
    a. None. Traffic matching this rule will not get logged.
    b. Log. Traffic matching this rule will get logged.
    c. Account. Traffic matching this rule will get stored in the accounting database for further investigation. Can be very resource consuming.
    d. Alert. Multitude of scripts that can be executed, syslogger etc.
    e. SnmpTrap. Sends SNMP traps.
    f. Mail. Sends mail using local sendmail on management module.
    g. UserDefined, like execution of customized scripts etc. This means that every time a packet matches this rule, the script etc will get executed. This can put quite a load on the logserver/management module.

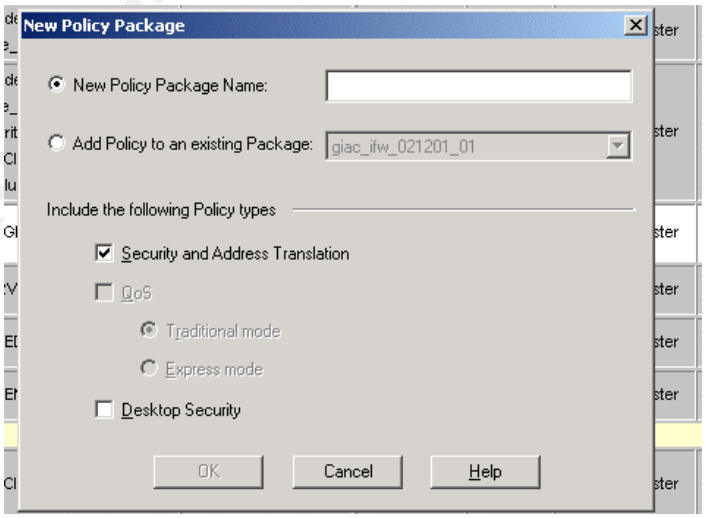
2.3.2. How to create a Rulebase from scratch (in NG)

Follow these steps:

- in case you have not done so, login to the SMARTDashboard (Policy Editor) with read/write privileges (depends on the useraccount and if somebody else is already using that policy editor in read/write mode)



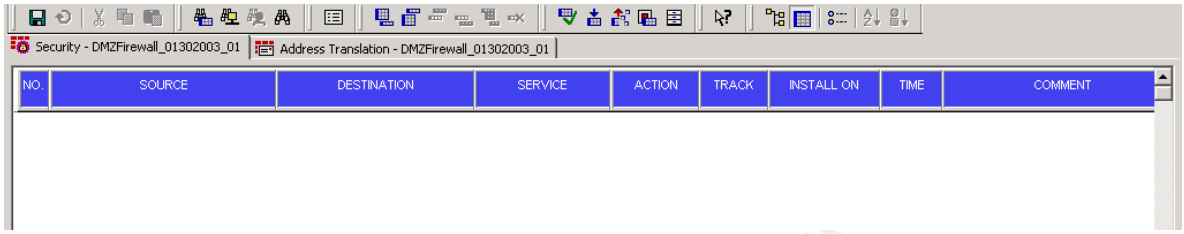
- create a new rulebase: click on File – New



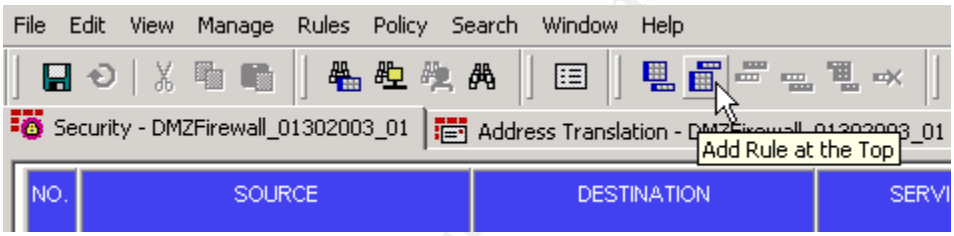
- check "Security and Address Translation" if you will need this in the policy later on

- give the package a name that makes sense, e.g. <name of firewall>_<timestamp in MMDDYYYY>_revision, like: DMZFirewall_01302003_01, that way you have a good overview over what rules you are dealing with. At the same time, you can keep track of it rather easily by looking at the timestamp and revision to see whether the policy is outdated or now

- hit OK, a new, clean rulebase appears

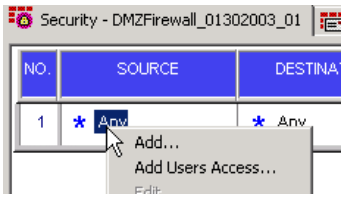


- add a new rule by e.g. clicking on the Add-Rule-At-Top-button

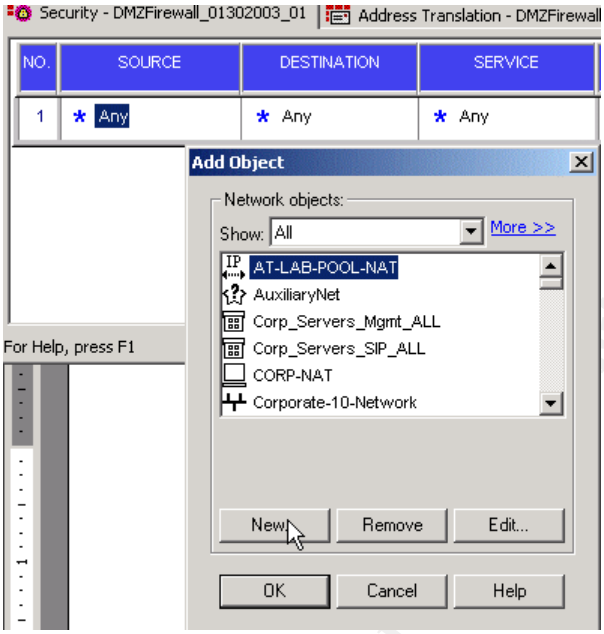


- a new rule is inserted. By default, all rules are added as default-drop-all

- add objects in the source and destination columns by right-clicking into each cell



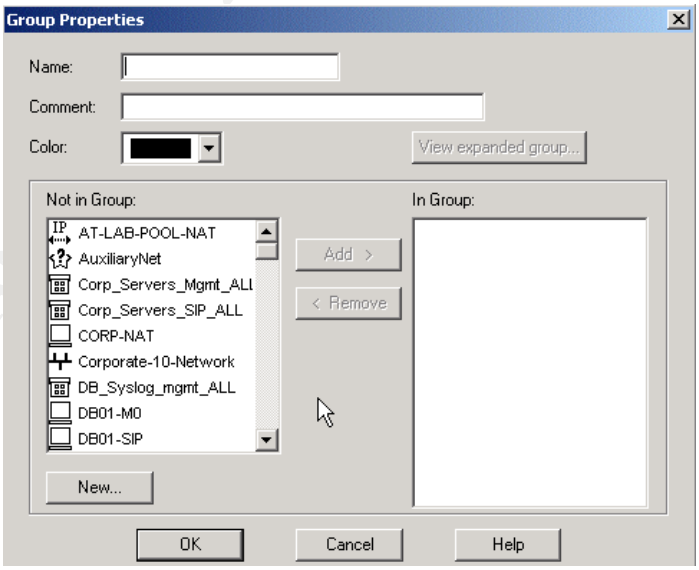
- left-click on e.g. "Add", that brings up a new window



- from here you can add either existing objects and groups, or create new objects and groups on the fly (new in NG, wasn't that easy in 4.1, good for workflow).

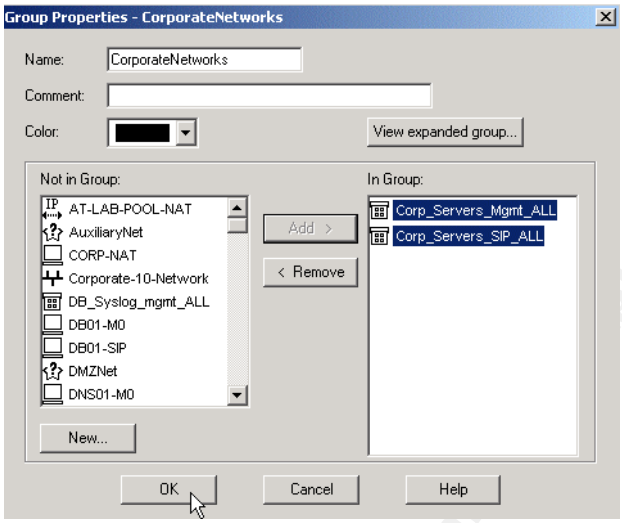
- adding services to the service column is similar.

- to create a group, in above step, click on New-Group-Simple Group, that brings up the "Group Window":



- in there, you assign a name to the group. Assign a name that makes sense and tells you about the contents

- select the objects you want to have in that group, and click "Add", it will move the objects you selected into the right column. Once you are done, click OK



- add it to the rulebase. by hitting OK once more

- add all the sources, destinations, services; select the type of action (e.g. accept) and whether you want to track it or not (off by default, you may want to change that)

- once you are done, a rule could look like this:

NO.	SOURCE	DESTINATION	SERVICE	ACTION	TRACK	INSTALL ON	TIME	COMMENT
1	CorporateNetworks	Inside_Mgmt_Networks	TCP ssh	accept	Log	* Policy Targets	* Any	management access

- you can leave the value in the Install-On column on the default (Policy Targets) if you desire, and select the actual firewalls on which this policy should be installed on later on when you start the policy installation process. I prefer to add the actual firewalls right here as well. I think it is easier for management purposes, and I can see faster which policies are installed on which firewalls

- you can add the firewalls by right-clicking into the field in the Install-On-column and select Add-Targets. Considering this, our rule would look like this:

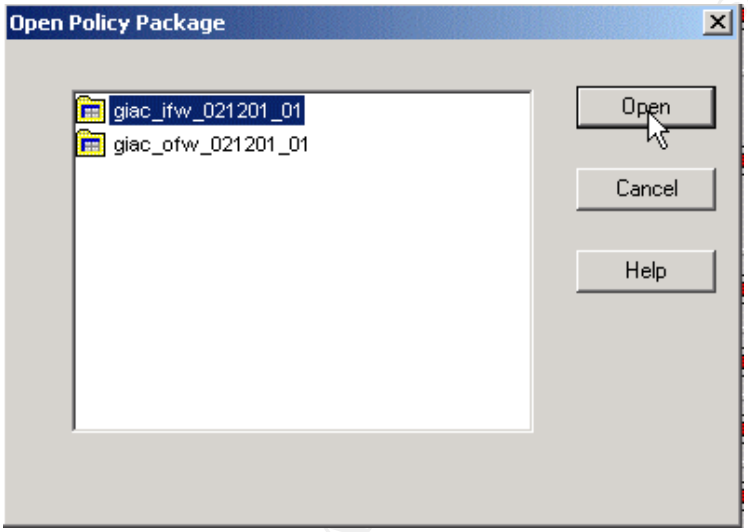
1	CorporateNetworks	Inside_Mgmt_Networks	TCP ssh	accept	Log	OFWCluster	* Any

- when you are done, make sure to save the policy (File- Save). If you want to install it, proceed to the next section

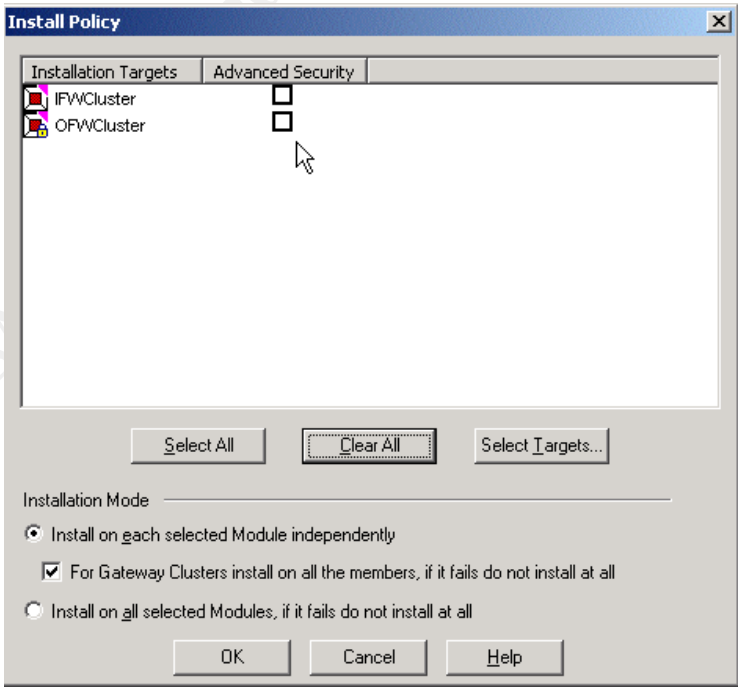
2.3.3. Installation of a Policy and Example for Troubleshooting

In order to install a new policy, the following has to be done:

- open the policy that should be installed

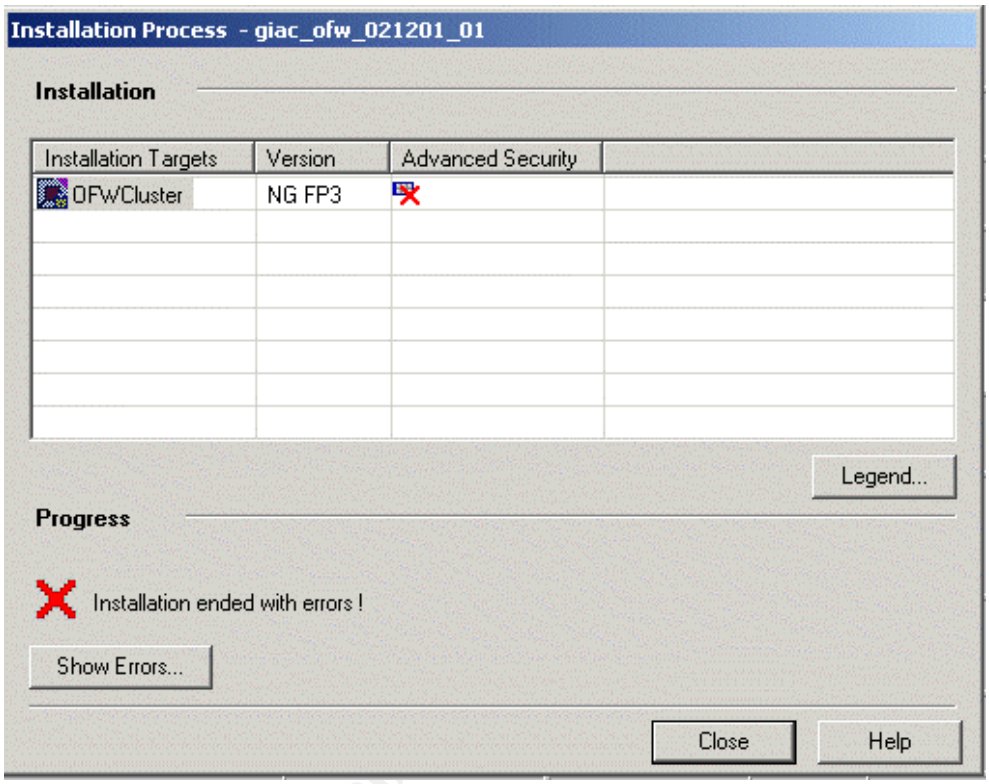


- push "install policy" and select the target firewalls that will run that policy, hit OK

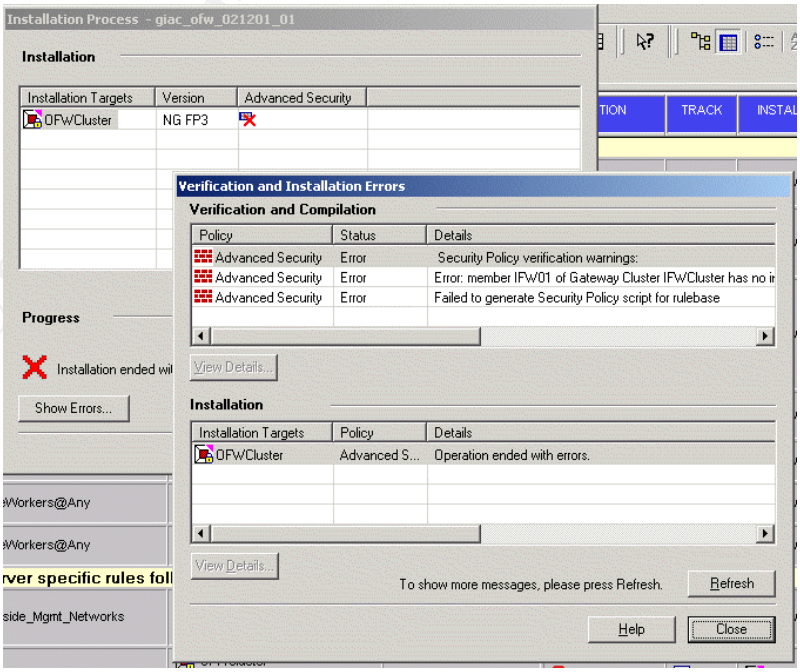


- this assumes that the firewalls are configured to be managed by the management module you are installing the policy from (maybe already running an existing policy that got installed before, from that same management module)

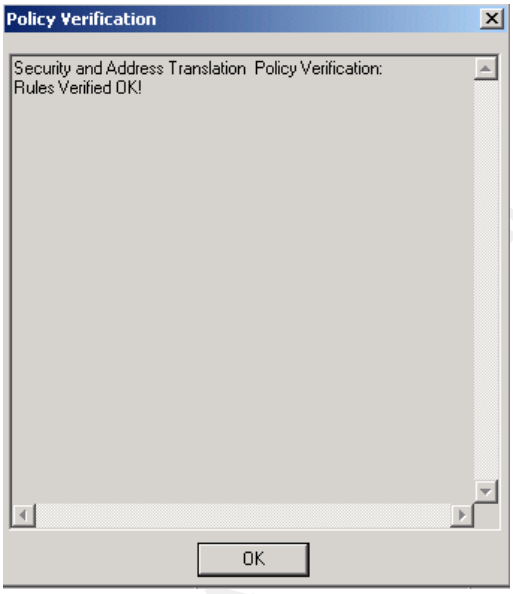
- if the policy installation fails:



- examine the error log / information the application generated, troubleshoot accordingly

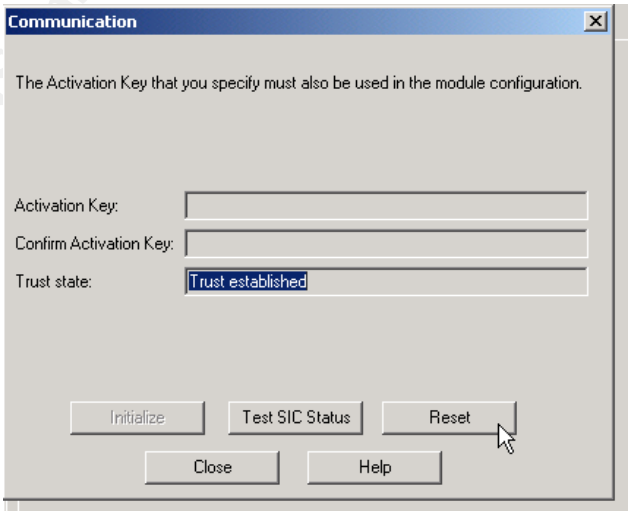


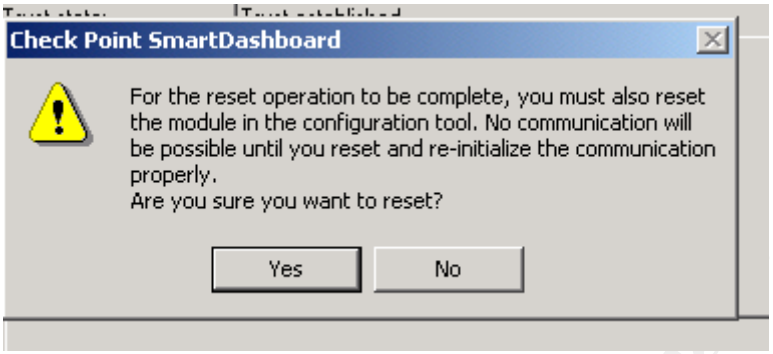
- If the policy is verified (by checking via VERIFY POLICY), there may be a communication issue between the firewall module and the management module. You may have to reset the SIC (Secure Internal Communication in FW-1 NG, the successor to the "putkeys" in FW-1 4.1 and earlier. Basic difference from experience: SIC is GUI-based, whereas the putkeys were command-line based



There are several locations where you can reset the SIC:

- in the policy, for a specific firewalls module
    - open up the Checkpoint object you want to reset the SIC for, click on Secure Internal Communication, reset SIC, type in new shared key (that's basically what SIC/putkeys are about, shared keys), hit Initialize. You will have to reset the SIC on the firewall module separately, and bounce the firewall module. You did not have to do this in 4.1, now you have to





- on the management server by resetting the SIC
  - this is global, it will affect the communication to all Checkpoint nodes that are controlled by this management module
  - In a root-shell on the management module, start cpconfig, run RESET INTERNAL CERTIFICATE AUTHORITY
  - confirm, done

```
[redacted]root# cpconfig
This program will let you re-configure
your SVN Foundation configuration.


Configuration Options:
----------------------
(1)  Licenses
(2)  Administrators
(3)  Management Clients
(4)  SNMP Extension
(5)  Random Pool
(6)  Certificate Authority
(7)  Certificate's Fingerprint

(8) Exit

Enter your choice (1-8) :6


Configuring Certificate Authority...
====================================


The FQDN (Fully Qualified Domain Name) of this Management Server
is required for proper operation of the Internal Certificate Authority.

Would you like to define it now (y/n) [y] ? y

You have already entered the FQDN of the Management Server.
Replacing the Management Server's FQDN will cause already generated ICA
certificates to contain an invalid CRL distribution point.
This might cause validation problems.
Internal CP entities will still be able to validate these certificates.
The FQDN of this Management Server is [redacted]
Do you want to change it (y/n) [n] ? y

Please enter the FQDN (Fully Qualified Domain Name) of this management:
```

- now you need to reset the SIC on each firewall module, then in the policy editor

```
This program will let you re-configure
your VPN-1 & FireWall-1 configuration.


Configuration Options:
----------------------
(1)  Licenses
(2)  SNMP Extension
(3)  PKCS#11 Token
(4)  Random Pool
(5)  Secure Internal Communication
(6)  Enable Check Point High Availability/State Synchronization
(7)  Automatic start of Check Point Products

(8) Exit

Enter your choice (1-8) :5


Configuring Secure Internal Communication...
============================================
The Secure Internal Communication is used for authentication between
Check Point components

Trust State: Trust established

Would you like re-initialize communication? (y/n) [n] ? y

Note: The Secure Internal Communication will be reset now.
No communication will be possible until you reset and
re-initialize the communication properly!
Are you sure? (y/n) [n] ?
```

2.3.4. Verifying the Rules in the Firewall Logs

We can verify that the firewall cluster is actually passing the traffic by looking at the logs on the firewall log server. For example, lets look at rule # 28 in the outside firewall cluster policy:

28	MX01-SIP-E0	INSIDEMX01-SIP	TCP smtp	accept	Log	OFWCluster	* Any

We execute the command `fw logswitch -ft` on the log server and see what we get:

```
14:02:00 accept OFWCluster >eth-s3p4c0 product: VPN-1 & FireWall-1;
src: MX01-SIP-E0; s_port: 55416; dst: INSIDEMX01-SIP; service: smtp;
proto: tcp; rule: 28;
```

It works, as you can see the log entry confirming it.

3. Assignment 3 - Audit

You have been asked to conduct a technical audit of GIAC's primary firewall in order to verify that the policies are correctly enforced as described in Assignments 1 and 2. To conduct the audit, you will need to:

- Plan the audit.
  - Describe the technical approach you will use to assess the firewall.
  - Be certain to include considerations such as what shift or day you would do the assessment.
  - Estimate costs and level of effort.
  - Identify risks and considerations and how they are addressed.
  - Remember the goal is to verify the firewall policy not perform a vulnerability assessment.

- Using the approach you described conduct the audit.
  - Demonstrate how you validated that the primary firewall is actually implementing GIAC Enterprise's security policy.
  - Be certain to include the tools and commands used. Include screen shots in your report if possible.
  - It is essential that you are actually verifying the firewall policy instead of auditing or vulnerability assessing other network devices.

- Evaluate the audit. Based on your assessment (and referring to data from your assessment):
  - Provide an analysis of the audit results.
  - Make recommendations for improvements or alternate architectures.
  - Supportive diagrams are strongly recommended for this part of the assignment.

Note: DO NOT simply submit the output of nmap or a similar tool here. It is fine to use any assessment tool you choose, but you must annotate/explain the output.

3.1. Planning the Audit

- Describe the technical approach you will use to assess the firewall.
- Be certain to include considerations such as what shift or day you would do the assessment.
- Estimate costs and level of effort.
- Identify risks and considerations and how they are addressed.
- Remember the goal is to verify the firewall policy not perform a vulnerability assessment.

3.1.1. Technical Approach

In order to verify the firewall policy of the primary firewall (the outside firewall cluster), we need to confirm that the communication in the network is indeed working as it is expected. In order do this, we will use the following approach:

- for a short period of time, replace each server that is mentioned in the source column of the outside firewall policy with the laptop sniffer
- configure the laptop with the server's IP address
- wherever possible, run a portscan against the destination(s) in the firewall ruleset, spanning all 65536 ports on both TCP and UDP. Sometimes this is not possible/appropriate (for example, when the destination is ANY and/or if one of the firewalls is the source, as there are no portscanners for the Nokia OS and it is not possible for me to compile any due to OS restrictions [no compiler for Nokia I had access to])
- test-connect to supposedly open ports
- for TCP ports: try to connect to the ports on the destination device(s) that are supposed to be open
- for UDP ports: try to verify the service
- at the same time, run a TCPDUMP on the firewall, on the interface that is supposed to handle the outgoing traffic to the specific destination(s) that are trying to get reached.

In order to make it easier to read the output, each rule of the outside firewall policy will be displayed in this section, together with the nmap, telnet or other commands that executed on the source-server. In addition to that, the TCPDUMP results from capturing the traffic on the firewall will be displayed afterwards.

The audit of each rule will have the following format:

- screenshot of the rule
- which interface is supposed to let the traffic out (after it has been processed by the rulebase)
- command that needs to get executed to generate the traffic and from which device this command needs to get executed
- tcpdump on the respective interface
- conclusion

Note: The tcpdump tool on Nokia boxes has a slightly different format. The basic format is:
```
11:44:24.496461 O 1.1.1.133 > 224.0.0.18:  VRRPv2-adver 20: vrid 3 pri
101
```
<TIMESTAMP> <DIRECTION> <SOURCE> <DESTINATION> <PACKET INFO>
(direction will be "O" for outgoing, "I" for incoming - towards the firewall from the outside)

3.1.2. Time of Day for doing the Audit

It is recommended to do the audit during a time window where the least amount of traffic is going through the GIAC network, i.e. the least amount of traffic being processed by the outside firewall cluster. Reason is that an extensive audit may put a significant load on the cluster. Also, the audit may crash the firewall cluster, thus impacting production / availability.

It was determined that the window for this audit should be in the early morning, 0500am to 0800am. Customers are most likely not too active during that time and GIAC employees do not start working before 0830am or so.

3.1.2.1. Day of Week

It was determined that the best day for doing the audit is after the week has started and before the week ends, i.e. Tuesday through Thursday. Also, admin personnel needs to be available, that is another big reason why we want to do that during the week and not on the weekends.

3.1.3. Costs and Level of Effort

Aside from the initial cost for the laptop (software is free), the man-hours need to be considered. Average fee for contracting work / security assessment should be around $250-$350 per hour per person. It is estimated that the whole procedure (setup, audit, evaluation of results, report) takes one full work week (5 business days), every workday being 8 hours.

The chunk of work will go into preparing everything and analysis of the results of the audit, as well as preparing documentation and recommendations, if they are needed. Doing the actual audit takes not as much time, since the tools are pretty much automated and just need to be fired off, then we can collect the results.

3.1.3.1. Preparations

The audit will be done by either replacing the servers with the scanner laptop or by adding the audit laptop into the network from where a rule needs to get tested out (if the ruleset allows testing from a general network, not from a specific server only). Replacing a server is not necessarily a desirable option because of possible production impact; it may have to be done though in order to fully test the firewalls.

Hence, where needed, we will need to ask the network admin/s to configure a port to be in certain VLANS ahead of time. The switchports would be shutdown, so nobody can take advantage of unused and fully configured switchports around the time of the audit. It is understood that the network admin/s needs to be available during the time of the audit in order to enable/disable ports when needed. After we are done with one network, the appropriate port will be disabled. After everything is done and we are satisfied with the results, the ports may be deactivated/unconfigured/moved out of the VLANs.

3.1.4. Risks and Considerations

Portscanning and establishing TCP connections via telnet etc will not do any harm to anything aside from creating log entries in the firewall logs as well as the syslogs of the exposed servers. The people responsible for the network equipment (routers/ switches/ firewalls) and the servers should be informed in time, to not cause any confusion when they see those log entries appearing in the log files. Also, all other employees, important suppliers and resellers will be informed of the audit ("maintenance window, may impact performance of the network"-kind of text). There will also be a message on the main GIAC/eKookie web site (same text, general "maintenance window"-style) to inform customers hitting the network during the time of the audit.

As soon as there are signs of serious business impact, the audit may have to get stopped and rescheduled.

However, it is not expected that anybody even experiences any slowness etc, since we are only checking the firewall policy, and not doing load-testing, vulnerability testing etc. To make sure that it we can recover as fast as possible in case of a device failure caused by the audit (and for liability reasons), all servers and devices will get fully backed up on the night before the audit.

3.1.4.1. Backup procedures before the Audit

Usually, every server is backed up incrementally every night, i.e. only changes are backed up. Before the audit however, the servers will get backed up fully (OS + application + data) to have a complete set of backups. The device configuration (switches/router) will get saved and stored on the backup server, as well as the firewall configurations. The firewall management server and the firewall logs will get backed up as well, to the backup server (TAPE01). All data on the TAPE server will then be backed up to tapes, to preserve it in case something may happen to the TAPE server itself (highly unlikely).

3.2. Conducting the Audit

3.2.1. Conventions

Not all TCPdump traces may be listed for space constraints, and not to bore the reader [grader]. Where appropriate, the references will be given instead (e.g. "tcpdump similar to 1.2.3 Rule 1 output").

3.2.2. Firewall Configuration Audit

Before we check the rulebase, we need to make sure that both firewalls in the cluster are hardened securitywise. There are 2 parts in this: the kernel version / system patch status and the system configuration. The relevant sections in the Nokia Voyager config from the firewalls and how they should be set is mentioned in brackets "[]" are as follows:

```
Network Access and Services
Access:
      Allow FTP access: disabled
      Allow telnet access: disabled
      Allow admin network login: disabled
      Allow com2 login: disabled
```

```
      Allow com3 login: disabled
Services:
      echo service: disabled
      discard service: disabled
      chargen service: disabled
      daytime service: disabled
      time service: disabled

Voyager Web Access
Access:
      Allow web access: enabled
SSL security:
      enabled 168 bits minimum

SSH (Secure Shell)
      SSH service: enabled
      Protocol version(s): 2
      Admin login: allowed [forbidden]
      Authentication modes: Password

SNMP Configuration
      snmpd: Disabled

NTP service:
      ntpd: enabled
```

Conclusion: It was found that all Nokia firewalls have the latest tested production kernel and packages installed. It was found that remote ssh as admin was enabled. This needs to be disabled. Firewall admins should only be able to access the Nokia box as standard user, then 'su' locally to admin.

3.2.3. Scan from the Outside

To start off, we will run a nmap scan of both TCP and UDP against the public network from the outside (in front of the outside firewall cluster), across all 65535 ports in both protocols.

```
bash# nmap –p 1-65535 -sT -P0 -T 3 1.1.1.128/25
bash# nmap –p 1-65535 -sU -P0 -T 3 1.1.1.128/25

All 65535 scanned ports on  (1.1.1.129) are: filtered.
All 65535 scanned ports on  (1.1.1.130) are: filtered.
All 65535 scanned ports on  (1.1.1.131) are: filtered.
Interesting ports on  (1.1.1.132):
(The 65532 ports scanned but not shown below are in state: filtered)
Port       State       Service
264/tcp    open        bgmp
265/tcp    open        maybeFW1
500/udp    open        isakmp

Interesting ports on  (1.1.1.133):
(The 65532 ports scanned but not shown below are in state: filtered)
Port       State       Service
264/tcp    open        bgmp
265/tcp    open        maybeFW1
500/udp    open        isakmp

Interesting ports on  (1.1.1.134):
(The 65532 ports scanned but not shown below are in state: filtered)
Port       State       Service
264/tcp    open        bgmp
```

```
265/tcp    open        maybeFW1
500/udp    open        isakmp

All 65535 scanned ports on  (1.1.1.145) are: filtered
All 65535 scanned ports on  (1.1.1.146) are: filtered
All 65535 scanned ports on  (1.1.1.147) are: filtered

Interesting ports on  (1.1.1.148):
(The 65534 ports scanned but not shown below are in state: filtered)
Port       State       Service
53/udp     open        domain-udp

Interesting ports on  (1.1.1.149):
(The 65534 ports scanned but not shown below are in state: filtered)
Port       State       Service
80/tcp     open        http
443/tcp    open        https


Interesting ports on  (1.1.1.151):
(The 65534 ports scanned but not shown below are in state: filtered)
Port       State       Service
25/tcp     open        smtp

bash# telnet 1.1.1.132 264
Trying 1.1.1.132...
Connected to 1.1.1.132.
Escape character is '^]'.

bash# telnet 1.1.1.132 265
Trying 1.1.1.132...
Connected to 1.1.1.132.
Escape character is '^]'.

bash# telnet 1.1.1.133 264
Trying 1.1.1.133...
Connected to 1.1.1.133.
Escape character is '^]'.

bash# telnet 1.1.1.133 265
Trying 1.1.1.133...
Connected to 1.1.1.133.
Escape character is '^]'.

bash# telnet 1.1.1.134 264
Trying 1.1.1.134...
Connected to 1.1.1.134.
Escape character is '^]'.

bash# telnet 1.1.1.134 265
Trying 1.1.1.134...
Connected to 1.1.1.134.
Escape character is '^]'.

bash# telnet www.ekookie.com 80
Trying 1.1.1.149...
Connected to www.ekookie.com.
Escape character is '^]'.
GET /index.html
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<HTML><HEAD>
<TITLE>Welcome to eKookie!</TITLE>
</HEAD><BODY>
[..]

bash# telnet 1.1.1.151 25
Trying 1.1.1.151...
Connected to mx.ekookie.com.
Escape character is '^]'.
220 MX01 ESMTP Postfix
QUIT
```

```
221 Bye
Connection closed by foreign host.

bash# dig @ns.ekookie.com www.ekookie.com a
[...]
;; got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 6
;; flags: qr rd ra; QUERY: 1, ANSWER: 4, AUTHORITY: 4, ADDITIONAL: 2
;; QUERY SECTION:
;;      www.ekookie.com, type = A, class = IN

;; ANSWER SECTION:
 [...]
www.ekookie.com.            17m44s IN A     1.1.1.149

bash# dig @ns.ekookie.com www.ibm.com a
; <<>> DiG 8.3 <<>> @ns.ekookie.com www.ibm.com a
; (1 server found)
;; res options: init recurs defnam dnsrch
;; got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 6
;; flags: qr rd ra; QUERY: 1, ANSWER: 4, AUTHORITY: 4, ADDITIONAL: 1
;; QUERY SECTION:
;;      www.ibm.com, type = A, class = IN

;; ANSWER SECTION:
www.ibm.com.            25m37s IN A     129.42.17.99
www.ibm.com.            25m37s IN A     129.42.18.99
www.ibm.com.            25m37s IN A     129.42.19.99
www.ibm.com.            25m37s IN A     129.42.16.99

;; AUTHORITY SECTION:
ibm.com.                9m55s IN NS     ns.watson.ibm.com.
ibm.com.                9m55s IN NS     ns.almaden.ibm.com.
ibm.com.                9m55s IN NS     internet-server.zurich.ibm.com.
ibm.com.                9m55s IN NS     ns.austin.ibm.com.

;; ADDITIONAL SECTION:
ns.watson.ibm.com.      9m55s IN A      129.34.20.80
```

(recursive DNS works as well)

```
bash# telnet www.ekookie.com 443
Trying 1.1.1.149...
Connected to www.ekookie.com.
Escape character is '^]'.
```

Results as expected: The router does not have any ports open, the firewall cluster has the usual ports open (264 for topology downloads, 265 for firewall gui-client connects – which are then accepted or denied by the firewall through policy, if somebody tries to connect to those ports, 500 for IKE/ISAKMP connections), DNS server responds on 53/udp (queries only, 53/tcp for zone transfers is restricted to pre-defined secondary DNS servers in the Internet/at the ISP, hence not showing), webserver responds on 80 and 443 for http and https connections, the ftp server does not respond, since the connections are restricted, the smtp relay responds on port 25/smtp. The dedicated NATs do not respond, since their usage is limited for internal-to-outside connections.

Conclusion: nothing major going on we would need to focus our immediate attention on. Lets move on, starting with auditing rule number 1 of the outside firewall policy.

3.2.4. Rule 1

To audit this rule, we do not need to generate any traffic by ourselves. Since the traffic originates from the firewall whenever VRRP is setup, we only need to listen in the actual traffic already being generated by the firewall clusters. Every cluster broadcasts VRRP information to vrrp.mcast.net, or 224.0.0.18. Note: Running NMAP was not an option, since "vrrp.mcast.net" is a multicast address, there will be no responses anyway.

interface traffic should be seen at:
eth-s1p1c0, the public interface

command:
none, listening into actual traffic

output:
```
tcpdump -i eth-s1p1c0 host 224.0.0.18
11:44:24.496461 O 1.1.1.133 > 224.0.0.18:  VRRPv2-adver 20: vrid 3 pri 101
11:44:24.867222 O 1.1.1.133 > 224.0.0.18:  VRRPv2-adver 20: vrid 1 pri 100
11:44:25.218382 O 1.1.1.133 > 224.0.0.18:  VRRPv2-adver 20: vrid 4 pri 101
[…]
```

Conclusion: works as expected.

3.2.5. Rule 2

ssh/ping/FW1 communication between the pair in the OFWCluster. We will execute separate commands to test each service that should be allowed. NMAP is not an option, since it is not supported for Nokia (see "Technical Approach").

interface this traffic should be seen at:
eth1c0, the heartbeat interface

command 1:
on the second firewall in the OFWCluster pair: `telnet 172.18.0.41 22` (22=ssh)
getting banner..
```
Connected to 172.18.0.2.
Escape character is '^]'.
SSH-2.0-OpenSSH_2.1.1
SSH-2.0-OpenSSH_2.1.1 [typed in]
$˜Îo|Þ-5W3£ÓÈ´Æ0diffie-hellman-group1-sha1ssh-dss)3des-cbc,blowfish-cbc,arcfour,cast128-
cbc)3des-cbc,blowfish-cbc,arcfour,cast128-cbc-hmac-sha1,hmac-md5,hmac-
ripemd160@openssh.com-hmac-sha1,hmac-md5,hmac-ripemd160@openssh.com       zlib,none
zlib,none
^]
```

output
```
bash# tcpdump -i eth1c0 port 22
```

```
12:11:55.195648 I 172.18.0.42.954 > 172.18.0.41.22: S 1418015116:1418015116(0) win 16384
<mss 512,nop,wscale 0,nop,nop,timestamp[|tcp]>
12:11:55.195882 O 172.18.0.41.22 > 172.18.0.42.954: S 1262699807:1262699807(0) ack
1418015117 win 16384 <mss 512,nop,wscale 0,nop,nop,timestamp[|tcp]>
12:11:55.196116 I 172.18.0.42.954 > 172.18.0.41.22: . ack 1 win 16384 <nop,nop,timestamp
1810449 1810278,nop,nop,[|tcp]>
12:11:55.197750 O 172.18.0.41.22 > 172.18.0.42.954: P 1:23(22) ack 1 win 16384
<nop,nop,timestamp 1810278 1810449,nop,nop,[|tcp]>
12:11:55.202286 I 172.18.0.42.954 > 172.18.0.41.22: . ack 23 win 16362 <nop,nop,timestamp
1810449 1810278,nop,nop,[|tcp]>
[…]
```

command 2:
on the second firewall in the OFWCluster pair: `telnet 172.18.0.41 256` (256=FW1 port)

output:

```
bash# tcpdump -i eth1c0 port 256
12:15:51.109096 I 172.18.0.42.2627 > 172.18.0.41.256: S 1464610643:1464610643(0) win
16384 <mss 512,nop,wscale 0,nop,nop,timestamp[|tcp]> [tos 0x10]
12:15:51.109266 O 172.18.0.41.256 > 172.18.0.42.2627: S 1309444849:1309444849(0) ack
1464610644 win 16384 <mss 512,nop,wscale 0,nop,nop,timestamp[|tcp]>
12:15:51.109452 I 172.18.0.42.2627 > 172.18.0.41.256: . ack 1 win 16384
<nop,nop,timestamp 1810921 1810750,nop,nop,[|tcp]> [tos 0x10]
```

command 3:
on the second firewall in the OFWCluster pair: ping 172.18.0.41

output:

```
bash# tcpdump -i eth1c0 icmp
12:21:55.077537 I 172.18.0.42 > 172.18.0.41: icmp: echo request
12:21:55.077736 O 172.18.0.41 > 172.18.0.42: icmp: echo reply
12:21:56.087792 I 172.18.0.42 > 172.18.0.41: icmp: echo request
12:21:56.087983 O 172.18.0.41 > 172.18.0.42: icmp: echo reply
```

Conclusion: all three services in that rule work as expected.

3.2.6. Rule 3

FWMGMT01 has the IP 172.18.1.20, we will configure our laptop with this IP and run nmap and telnet against the firewall cluster, OFW01 and OFW02.

firewall interface we expect this traffic at:
eth2c0, the external network device management interface, on OFW01 and OFW02

command 1:
from the laptop, `telnet 172.18.0.2 18192` (CPD_amon port)

output:

```
bash# tcpdump -i eth2c0 host 172.18.1.20
tcpdump: listening on eth-s1p1c0
12:57:52.119832 I 172.18.1.20.35934 > 172.18.0.2.18192: S 2385020403:2385020403(0) win
24820 <nop,nop,sackOK,mss 1460> (DF)
```

```
12:57:52.120196 O 172.18.0.2.18192 > 172.18.1.20.35934: S 3152571953:3152571953(0) ack
2385020404 win 16384 <mss 512>
```

Conclusion: works. Same for OFW02 / 172.18.0.3 (not listed for space reasons)

command 2:
from the laptop, with FWMGMT01's IP address: `telnet 172.18.0.2 256` (256=FW1)
output:

```
bash# tcpdump -i eth2c0 host 172.18.1.20 and port 256
13:42:40.136988 I 172.18.1.20.36349 > 172.18.0.2.256: S 1440108583:1440108583(0) win
24820 <nop,nop,sackOK,mss 1460> (DF)
13:42:40.137289 O 172.18.0.2.256 > 172.18.1.20.36349: S 3664335194:3664335194(0) ack
1440108584 win 16384 <mss 512>
13:42:40.137732 I 172.18.1.20.36349 > 172.18.0.2.256: . ack 1 win 25088 (DF)
13:42:42.248174 I 172.18.1.20.36349 > 172.18.0.2.256: F 1:1(0) ack 1 win 25088 (DF)
13:42:42.248326 O 172.18.0.2.256 > 172.18.1.20.36349: . ack 2 win 16384
```

Conclusion: works (same for OFW02)

command 3:
from the laptop, `telnet 172.18.0.2 22` (22=ssh), getting banner..

```
Connected to 172.18.0.2.
Escape character is '^]'.
SSH-2.0-OpenSSH_2.1.1
SSH-2.0-OpenSSH_2.1.1 [typed in]
$˜Îo|Þ-5W3£ÓÈ´Æ0diffie-hellman-group1-sha1ssh-dss)3des-cbc,blowfish-cbc,arcfour,cast128-
cbc)3des-cbc,blowfish-cbc,arcfour,cast128-cbc-hmac-sha1,hmac-md5,hmac-
ripemd160@openssh.com-hmac-sha1,hmac-md5,hmac-ripemd160@openssh.com       zlib,none
zlib,none
^]
```

output:

```
bash# tcpdump -i eth2c0 port 22
13:56:32.808399 I 172.18.1.20.49111 > 172.18.0.2.22: S 130691030:130691030(0) win 24820
<nop,nop,sackOK,mss 1460> (DF)
13:56:32.808858 O 172.18.0.2.22 > 172.18.1.20.49111: S 898074621:898074621(0) ack
130691031 win 16384 <mss 512>
13:56:32.809120 I 172.18.1.20.49111 > 172.18.0.2.22: . ack 1 win 25088 (DF)
13:56:32.811856 O 172.18.0.2.22 > 172.18.1.20.49111: P 1:23(22) ack 1 win 16384
13:56:32.812085 I 172.18.1.20.49111 > 172.18.0.2.22: . ack 23 win 25088 (DF)
13:56:46.131630 I 172.18.1.20.49111 > 172.18.0.2.22: P 1:24(23) ack 23 win 25088 (DF)
13:56:46.132196 O 172.18.0.2.22 > 172.18.1.20.49111: P 23:319(296) ack 24 win 16384
13:56:46.229918 I 172.18.1.20.49111 > 172.18.0.2.22: . ack 319 win 25088 (DF)
13:56:47.472395 I 172.18.1.20.49111 > 172.18.0.2.22: P 24:26(2) ack 319 win 25088 (DF)
13:56:47.582580 O 172.18.0.2.22 > 172.18.1.20.49111: . ack 26 win 16384
13:56:48.742180 I 172.18.1.20.49111 > 172.18.0.2.22: F 26:26(0) ack 319 win 25088 (DF)
13:56:48.742392 O 172.18.0.2.22 > 172.18.1.20.49111: . ack 27 win 16384
13:56:48.743128 O 172.18.0.2.22 > 172.18.1.20.49111: F 319:319(0) ack 27 win 16384
13:56:48.743343 I 172.18.1.20.49111 > 172.18.0.2.22: . ack 320 win 25088 (DF)
```

Conclusion: works as expected (same for OFW02).

command 4:
from the laptop, `telnet 172.18.0.2 18264` (18264=FW1_ica_services)

output:

```
bash# tcpdump -i eth2c0 host 172.18.1.20 and port 18264
14:08:51.398533 I 172.18.1.20.36398 > 172.18.0.2.18264: S 1282077765:1282077765(0) win
24820 <nop,nop,sackOK,mss 1460> (DF)
14:08:51.408135 O 172.18.0.2.10001 > 172.18.1.20.18264: S 1282077765:1282077765(0) win
24820 <nop,nop,sackOK,mss 1460> (DF)
14:08:51.409059 I 172.18.1.20.18264 > 172.18.0.2.10001: S 1485893042:1485893042(0) ack
1282077766 win 24820 <nop,nop,sackOK,mss 1460> (DF)
```

```
14:08:51.409179 O 172.18.0.2.18264 > 172.18.1.20.36398: S 1485893042:1485893042(0) ack
1282077766 win 24820 <nop,nop,sackOK,mss 1460> (DF)
14:08:51.409594 I 172.18.1.20.36398 > 172.18.0.2.18264: . ack 1 win 24820 (DF)
14:08:51.409775 O 172.18.0.2.10001 > 172.18.1.20.18264: . ack 1 win 24820 (DF)
```

Conclusion: works as expected (same for OFW02).

command 5:
run nmap against the firewall cluster (OFW01 and OFW02, here only OFW01 is shown, same output for OFW02)

```
bash# nmap -sS -P0 -p 1-65535 172.18.0.2
```

and

```
bash# nmap -sU -P0 -p 1-65535 172.18.0.2
18192, 18264
Interesting ports on  (172.18.0.2):
(The 65532 ports scanned but not shown below are in state: filtered)
Port       State       Service
22/tcp     open        ssh
264/tcp    open        bgmp
18192/tcp  open        unknown
18264/tcp  open        unknown

UDP scan (abbrev.):
500/udp    open        isakmp
```

3.2.7. Rule 4

4	OFW_ALL	FWMGMT01	TCP FW1_log TCP FW1	accept	Log	OFWCluster	* Any	- fw traffic logging to management server - FW1 protocol for fetching policy upon startup

Logging to- and pulling policy from- FWMGMT01. FW1_log: 257/tcp, FW1: 256.

firewall interface we expect this traffic at:
eth2c0, the external network device management interface, on OFW01 and OFW02

command 1:
from the first firewall in the pair: `telnet 172.18.1.20 257`

output:

```
tcpdump -i eth-s2p4c0 port 257
14:40:17.739338 O 172.18.0.2.3061 > 172.18.1.20.257: S 3166499820:3166499820(0) win 16384
<mss 512,nop,wscale 0,nop,nop,timestamp[|tcp]> [tos 0x10]
14:40:17.739599 I 172.18.1.20.257 > 172.18.0.2.3061: S 3222990061:3222990061(0) ack
3166499821 win 25000 <nop,nop,timestamp 782746085 1828233,nop,[|tcp]> (DF)
14:40:17.739733 O 172.18.0.2.3061 > 172.18.1.20.257: . ack 1 win 16384 <nop,nop,timestamp
1828233 782746085> [tos 0x10]
```

As you can see, connection gets established.

Similar results for the FW1 service on port 256, from both firewalls.

Conclusion: this rule works as expected.

3.2.8. Rule 5

For site-to-site VPNs, VPN client connections and topology downloads. IKE: 500/udp, ESP: IP protocol 50, FW1_topo: 264/tcp.

Interface:
according to the rule, this is expected to work on every interface of each firewall, especially from the Internet. Here, we capture eth-s1p1c0, the public interface.

For this test, the laptop will be assigned a public IP from VLAN 22, the router-to-firewall VLAN. The IP assigned is 1.1.1.135.

We will use 3 NMAP scans:

```
bash# nmap -sS -p 1-65535 -P0 1.1.1.132
bash# nmap -sU -p 1-65535 -P0 1.1.1.132
bash# nmap -sO -P0 1.1.1.132
```

(IP protocol scan, to see whether ESP shows up)

Also, "telnet" was used to verify that the TCP connection to port 264 (FW1_topo service) is getting established. This was confirmed (TCP dump not listed, as no different to others above). The scans will be against the VRRP address of the cluster, as this is represented by the "OFWCluster" object in the rule.

```
Interesting ports on  (1.1.1.132):
(The 44997 ports scanned but not shown below are in state: filtered)
Port       State       Service
264/tcp  open        bgmp

UDP scan:
500/udp  open        isakmp

Protocol scan:
[..]
48         open        mhrp
49         open        bna
50         open        esp
51         open        ah
52         open        i-nlsp
[..]
```

As you can see, IP proto 50, ESP is supported.

How would we generate ESP traffic? We could use HPING [HPING].

The command for generating a packet that uses IP protocol 50, we would type:

```
bash# hping --rawip -H 50 1.1.1.132

output in tcpdump:
15:43:29.729766 I 1.1.1.3 > 1.1.1.132: ESP(spi=55555555,seq=0x55555555) (DF)
15:43:30.727500 I 1.1.1.3 > 1.1.1.132: ESP(spi=55555555,seq=0x55555555) (DF)
15:43:31.727496 I 1.1.1.3 > 1.1.1.132: ESP(spi=55555555,seq=0x55555555) (DF)
15:43:32.727547 I 1.1.1.3 > 1.1.1.132: ESP(spi=55555555,seq=0x55555555) (DF)
```

However, this does not prove that ESP is actually accepted in the firewall, as this packet is not valid and will not elicit a response (instead, it will get dropped by the firewall).

In order to verify that ESP is working, we listen in into actual VPN communication between e.g. a supplier and the GIAC network, here SupplierA, whose VPN gateway has the IP 5.3.4.2

interface:
eth-s1p1c0, the public interface of the firewall.

output:

```
16:05:43.857921 I 5.3.4.2 > 1.1.1.132: ESP(spi=9fcaa91e,seq=0x1d) (ttl 254, id 55403)
16:05:43.859010 O 1.1.1.132 > 5.3.4.2: ESP(spi=30a0e3a9,seq=0x18) (ttl 255, id 3739)
16:05:43.859852 I 5.3.4.2 > 1.1.1.132: ESP(spi=9fcaa91e,seq=0x1e) (ttl 254, id 55404)
```

This looks like communication would be working, but we cannot tell it from the packet contents (sequence numbers etc are not helpful here, since the packets are encrypted and the SPI and the id are changing with every packet.

One more place where we can check this are the FW-1 logs themselves.

E.g.

```
16:04:06 encrypt OFW01  >daemon proto tcp src SupplierA_VPNGateway dst FTP01-SIP-E0
service ftp s_port 40894 dstkeyid 0x9fcaa91e rule 10 scheme: IKE methods: Combined ESP:
3DES + SHA1 + PFS xlatesrc SupplierA_DB xlatedst FTP01-SIP-E0 xlatesport 40894 xlatedport
ftp
```

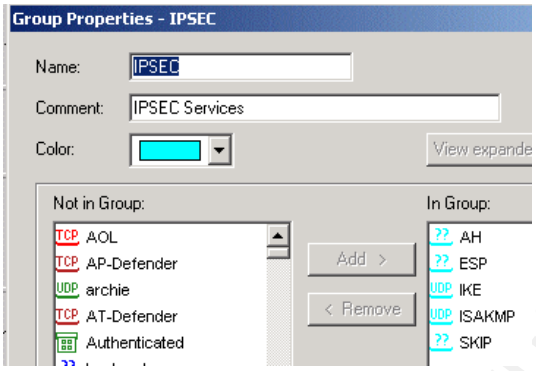
This means that packets are getting encrypted/decrypted at the OFWCluster, hence, ESP works (as well as IKE, for that matter).

3.2.9. Rules 6 through 10

6	OFWCluster ReselleA_VPNGateway	OFWCluster ReselleA_VPNGateway	IPSEC	accept	Log	OFWCluster	* Any	IKE/ESP/AH for site-to-site VPN
7	OFWCluster ReselleB_VPNGateway	OFWCluster ReselleB_VPNGateway	IPSEC	accept	Log	OFWCluster	* Any	IKE/ESP/AH for site-to-site VPN
8	OFWCluster SupplierA_VPNGateway	OFWCluster SupplierA_VPNGateway	IPSEC	accept	Log	OFWCluster	* Any	IKE/ESP/AH for site-to-site VPN
9	OFWCluster SupplierB_VPNGateway	OFWCluster SupplierB_VPNGateway	IPSEC	accept	Log	OFWCluster	* Any	IKE/ESP/AH for site-to-site VPN
10	Reseller_Databases_ALL Supplier_Databases_ALL	FTP01-SIP-E0	TCP ftp	Encrypt	Log	OFWCluster	* Any	- Resellers: download from FTP site - Suppliers: upload to FTP site all via VPN tunnels, encrypted.

These rules are grouped together because 6-9 are basically the same (site-to-site VPN rules with the suppliers and resellers) and 10 is the application of 6-9 (why they are needed): site-to-site VPNs in order to facilitate the FTP transfers between all suppliers, resellers on the one side and GIAC on the other side.

However, if we doublecheck what is contained in the IPSEC service group, we see:



AH, ESP, IKE, ISAKMP, SKIP are all in the IPSEC group. Of these are redundant: AH (not supported by NG), ISAKMP and SKIP (both are not used). So the only members of this group (and hence only allowed by the OFWCluster) are ESP and IKE.

We can see that there is an "already covered in a previous rule"-issue between rule 5 and the rules 6-9, since rule 5 covers the suppliers and resellers with ANY in the source column already. FW-1 does not notify us of that since 6-9 contain both our VPN endpoint and the suppliers/resellers in both columns. This does not disrupt any service or have any impact whatsoever. The only thing to consider is that traffic actually may get passed by a different rule than expected, thus making it harder to troubleshoot in the logs. For clarification purposes, I would recommend moving these rules before rule 5, so it would be more clear where to look if it needs to be checked out.

One way to test whether the suppliers and resellers are able to actually FTP to/from the FTP server through the VPN is to check the FW-1 logfiles as well.

This is an excerpt from the FW-1 NG logfile (we asked on of the suppliers to test-FTP during the audit to be able to get this log entry):

```
Number:           1321403
Date:             28Jan2003
Time:             16:34:56
Product:          VPN-1 & FireWall-1
Interface:        eth-s1p1c0
Origin:           OFW01
Type:             Log
Action:           Decrypt
Service:          ftp
Source:           SupplierA_DB
Destination:      FTP01-SIP-E0
Protocol:         tcp
Rule:             10
Source Port:      41309
Source Key ID:    0x9fcaa91e
Encryption Scheme:  IKE
VPN Peer Gateway:   SupplierA_Gateway
Encryption Methods: ESP: 3DES + SHA1 + PFS
```

This confirms that the firewall decrypts the packet.

How would we see whether FTP works though, considering the ports 21 and 20 are both used, how do we know that the FTP session itself works?

Answer: by doing a tcpdump on the FTP server while an FTP session is getting established and sample data is transferred (for example, the results of an "ls"):

```
bash# tcpdump -n -i qfe0 port 20 or port 21
16:55:19.802548 5.3.4.7.3917 > 1.1.1.150.21: S 3619486943:3619486943(0) win 16384 <mss
1460,nop,nop,sackOK> (DF)
16:55:19.802851 1.1.1.150.21 > 5.3.4.7.3917: S 3205698520:3205698520(0) ack 3619486944
win 65535 <mss 1460>
16:55:19.804729 5.3.4.7.3917 > 1.1.1.150.21: . ack 1 win 17520 (DF)
16:55:19.899286 1.1.1.150.21 > 5.3.4.7.3917: P 1:50(49) ack 1 win 65535 (DF) [tos 0x10]
16:55:20.058457 5.3.4.7.3917 > 1.1.1.150.21: . ack 50 win 17471 (DF)
16:55:21.216927 5.3.4.7.3917 > 1.1.1.150.21: P 1:11(10) ack 50 win 17471 (DF)
16:55:21.218833 1.1.1.150.21 > 5.3.4.7.3917: P 50:82(32) ack 11 win 65535 (DF) [tos 0x10]
16:55:21.360541 5.3.4.7.3917 > 1.1.1.150.21: . ack 82 win 17439 (DF)
16:55:22.785386 5.3.4.7.3917 > 1.1.1.150.21: P 11:24(13) ack 82 win 17439 (DF)
16:55:22.811474 1.1.1.150.21 > 5.3.4.7.3917: P 82:107(25) ack 24 win 65535 (DF) [tos
0x10]
16:55:22.963053 5.3.4.7.3917 > 1.1.1.150.21: . ack 107 win 17414 (DF)
16:55:24.749500 5.3.4.7.3917 > 1.1.1.150.21: P 24:49(25) ack 107 win 17414 (DF)
16:55:24.749974 1.1.1.150.21 > 5.3.4.7.3917: P 107:137(30) ack 49 win 65535 (DF) [tos
0x10]
16:55:24.752876 5.3.4.7.3917 > 1.1.1.150.21: P 49:55(6) ack 137 win 17384 (DF)
16:55:24.753818 1.1.1.150.20 > 5.3.4.7.3918: S 3438221220:3438221220(0) win 65535 <mss
1460,nop,wscale 1,nop,nop,timestamp 27010917 0> (DF) [tos 0x8]
16:55:24.755747 5.3.4.7.3918 > 1.1.1.150.20: S 3620760566:3620760566(0) ack 3438221221
win 17520 <mss 1460,nop,wscale 0,nop,nop,timestamp 0 0> (DF)
16:55:24.756016 1.1.1.150.20 > 5.3.4.7.3918: . ack 1 win 33304 <nop,nop,timestamp
27010917 0> (DF) [tos 0x8]
```

We can see how 5.3.4.7 (SupplierA_DB) and 1.1.1.150 (FTP01 service IP) have a connection going to/from port 21 of the FTP server, then at some point FTP01 initiates a connection FROM port 20 TO port 3918 of the SupplierA_DB server. This transferrs the actual data (while 21/tcp is only for ftp commands). Similar testing is done with all other resellers and suppliers to confirm that their VPN access works as well.

Conclusion: The VPNs as well as FTP through the VPN are working.

3.2.10. Rules 11 through 17

	VPN Client specific rules follow						
11	SystemAdministrators@Any	Public_Servers_Mgmt_ALL Inside_ProdServers_Mgmt_ALL	TCP ssh ping	Client Encrypt	Log	OFWCluster	
12	NetworkAdministrators@Any	Outside_Device_Mgmt_ALL Inside_Device_Mgmt_ALL	TCP ssh ping	Client Encrypt	Log	OFWCluster	
13	SecurityAdministrators@Any	Outside_Mgmt_Networks Inside_ProdServers_Mgmt_ALL Security_Servers_Mgmt_ALL OFWCluster IFWCluster	ping TCP ssh	Client Encrypt	Log	OFWCluster	
14	SecurityAdministrators@Any	FWMGMT01	TCP CPMI TCP FW1_mgmt	Client Encrypt	Log	OFWCluster	
15	AllAdmins@Any	TSERVER01-M0	TCP TerminalServerPorts	Client Encrypt	Log	OFWCluster	
16	TeleWorkers@Any	INSIDEDNS01-SIP	UDP domain-udp	Client Encrypt	Log	OFWCluster	
17	TeleWorkers@Any	INSIDEMX01-SIP	TCP smtp	Client Encrypt	Log	OFWCluster	
14	SecurityAdministrators@Any	FWMGMT01	TCP CPMI TCP FW1_mgmt	Client Encrypt	Log	OFWCluster	
15	AllAdmins@Any	TSERVER01-M0	TCP TerminalServerPorts	Client Encrypt	Log	OFWCluster	
16	TeleWorkers@Any	INSIDEDNS01-SIP	UDP domain-udp	Client Encrypt	Log	OFWCluster	
17	TeleWorkers@Any	INSIDEMX01-SIP	TCP smtp	Client Encrypt	Log	OFWCluster	

What we expect:

1) VPN user authenticates to OFWCluster, gets assigned an NAT IP address out of the IP pool for VPN client connections (172.18.0.64/27)
2) this is transparent to the client: the client only sees its IP talking to the original destination IP and vice versa. The destination device only sees the NAT as source IP and responds to that.
3) the place where we can see the translation happening is in the FW-1 logs of the outside firewall cluster. TCPdump on the firewall cluster will only show the original source IP as well as the destination IP.

I will not prove this for all rules listed above, out of space constraints. I will rather pick one rule as a proof of concept, in order to show the basic functionality. I picked rule number 14 since it also introduces new services, CPMI (18190/tcp) and FW1_mgmt (258/tcp). Both are for remote management of the FW-1 management server.

3.2.11. Rule 14

14	SecurityAdministrators@Any	FWMGMT01	TCP CPMI TCP FW1_mgmt	Client Encrypt	Log	OFWCluster	* Any	firewall management through Client VPN

1) user coming from the Internet with the IP 2.0.2.2 (IANA-unassigned IP, here as sample IP for an user coming through an ISP) and using the VPN client authenticates against the outside firewall cluster (user "sol" is a Security Administrator) and gets assigned an IP out of the IP pool (here 172.18.0.65)

```
17:43:32 authcrypt OFW01 >daemon src 2.0.2.2 user sol rule 0 reason Client Encryption:
Authenticated by SecurID scheme: IKE methods: 3DES,IKE,SHA1
```

2) user fires up his FW-1 mgmt client, like the policy / rulebase editor ("SmartDashboard"), one can see the packets crossing OFW01:

```
17:43:32 decrypt OFW01 >daemon proto tcp src 2.0.2.2 dst FWMGMT01 service CPMI s_port
4053 srckeyid 0xc6a5c8ec rule 14 user sol scheme: IKE methods: Combined ESP: 3DES +
SHA1 xlatesrc 172.18.0.65 xlatedst FWMGMT01 xlatesport 4053 xlatedport CPMI
```

The original source IP is 2.0.2.2, the original destination IP is 172.18.1.20 (or FWMGMT01 since it resolved in the FW-1 log viewer), the NAT IP assigned to 2.0.2.2 is 172.18.0.65 (xlatesrc). Source and destination port as well as the destination IP of FWMGMT01 stay unchanged, they will not get translated.

In order to see this on a packet level during monitoring, I will use "fw monitor" on the firewall as it gives more in-depth view on what is happening. It also watches ALL interfaces, so we do not have to run two tcpdumps at the same time etc; fw monitor is more suitable to get the big picture in a situation like this.

We run fw monitor like this on the firewall:

```
fw monitor -e "accept (dst=172.18.1.20 and dport=18190) or (src=172.18.1.20 and
sport=18190);"
```

I.e. watch all traffic to and from FWMGMT01 that relates to the management port, CPMI – 18190/tcp.

We see the following in this session:

```
eth-s1p1c0:i: 2.0.2.2 -> 172.18.1.20 (TCP) len=48 id=233 TCP: 4064 -> 18190 .S....
seq=10a56779 ack=00000000
eth-s1p1c0:I: 2.0.2.2 -> 172.18.1.20 (TCP) len=48 id=233 TCP: 4064 -> 18190 .S....
seq=10a56779 ack=00000000

eth-s1p4c0:o: 2.0.2.2 -> 172.18.1.20 (TCP) len=48 id=233 TCP: 4064 -> 18190 .S....
seq=10a56779 ack=00000000
eth-s1p4c0:O: 172.18.0.65 -> 172.18.1.20 (TCP) len=48 id=233 TCP: 4064 -> 18190 .S....
seq=10a56779 ack=00000000
eth-s1p4c0:i: 172.18.1.20 -> 172.18.0.65 (TCP) len=48 id=26730 TCP: 18190 -> 4064
.S..A. seq=a3de00c9 ack=10a5677a
eth-s1p4c0:I: 172.18.1.20 -> 2.0.2.2 (TCP) len=48 id=26730 TCP: 18190 -> 4064 .S..A.
seq=a3de00c9 ack=10a5677a

eth-s1p1c0:o: 172.18.1.20 -> 2.0.2.2 (TCP) len=48 id=26730 TCP: 18190 -> 4064 .S..A.
seq=a3de00c9 ack=10a5677a
```

Legend:
"i" = incoming before rulebase processing
"I" = incoming after rulebase processing

"o" = outgoing before network address translation
"O"= outgoing after network address translation

One can see how the original packet comes in through eth-s1p1c0, the outside interface, gets processed, forwarded to eth-s1p4c0, the inside connection to the management server, then translated into the VPN pool IP NAT 172.18.0.65, then sent out interface eth-s1p4c0.

The management server sees a request coming from the NAT and responds to it:

```
bash# tcpdump port 18190
18:41:21.886315 172.18.0.65.4089 > FWMGMT01.18190: R 796512963:796512963(0) win 0
18:41:33.788551 172.18.0.65.4097 > FWMGMT01.18190: S 920826052:920826052(0) win 16384
<mss 1460,nop,nop,sackOK>
18:41:33.788579 FWMGMT01.18190 > 172.18.0.65.4097: S 3065224774:3065224774(0) ack
920826053 win 24820 <nop,nop,sackOK,mss 1460> (DF)
18:41:33.801969 172.18.0.65.4097 > FWMGMT01.18190: . ack 1 win 17520
18:41:33.802607 172.18.0.65.4097 > FWMGMT01.18190: P 1:5(4) ack 1 win 17520
18:41:33.802660 FWMGMT01.18190 > 172.18.0.65.4097: . ack 5 win 24820 (DF)
18:41:33.802950 FWMGMT01.18190 > 172.18.0.65.4097: P 1:5(4) ack 5 win 24820 (DF)
```

Similar traces are obtained for the FW1_mgmt service. The connection via the VPN client was also tested from different locations, with the same result.

The other rules listed above (11,12,13 and 15,16,17) were also tested in a similar fashion, same success: all VPN client rules are working as expected.

NMAP scans (1-65535 tcp and udp) did not reveal anything unusual, all other ports were found closed when scanned from the laptop using the VPN client.

3.2.12. Rule 18

Since this rule has a network scope (no specific IPs as source, more a whole network), we will not need to replace a server/device to test this rule. We just assign an unused IP to our laptop, from those networks. In this case 172.18.0.24 from the VLAN 400 (public server management).

interface: eth-s1p3c0, public server management.

command:
on the laptop: `ping 172.18.0.17` (VRRP IP of OFWCluster in that network)

output:

```
bash# tcpdump -i eth-s1p3c0 icmp
12:21:55.077537 I 172.18.0.24 > 172.18.0.17: icmp: echo request
12:21:55.077736 O 172.18.0.17 > 172.18.0.24: icmp: echo reply
12:21:56.087792 I 172.18.0.24 > 172.18.0.17: icmp: echo request
12:21:56.087983 O 172.18.0.17 > 172.18.0.24: icmp: echo reply
```

Conclusion: works as expected.

Recommendation: The source column should be further restricted to only allow actually registered servers to be accepted under this rule. Details will be stated at the end of this assignment.

3.2.13. Rule 19

The stealth rule drops all other traffic headed for the outside firewall cluster that was not allowed in a previous rule.

To demonstrate this rule, we keep the laptop in the network from the previous rule, with the same IP address (172.18.0.24) and try to hit ports that are not allowed by any of the previous rules, such as connecting to the telnet port (23/tcp) of OFW01.

interface: eth-s1p3c0, public server management.

command:
on the laptop: `telnet 172.18.0.18` (OFW01 in that network)

output:

```
bash# tcpdump -i eth-s1p3c0 port 23
19:44:44.395493 I 172.18.0.24.34991 > 172.18.0.18.23: S 1479496962:1479496962(0) win
24820 <nop,nop,sackOK,mss 1460> (DF)
19:44:47.759303 I 172.18.0.24.34991 > 172.18.0.18.23: S 1479496962:1479496962(0) win
24820 <nop,nop,sackOK,mss 1460> (DF)
19:44:54.509173 I 172.18.0.24.34991 > 172.18.0.18.23: S 1479496962:1479496962(0) win
24820 <nop,nop,sackOK,mss 1460> (DF)
```

Conclusion: the traffic arrives at the firewall, but gets dropped. Otherwise, one would see a response coming from the firewall itself.

If the traffic would have been allowed, but the port actually not be open on the firewall, the tcpdump would have looked like this:

```
19:49:03.072805 I 172.18.0.24.63306 > 172.18.0.18.23: S 2423496499:2423496499(0) win 8760
<mss 1460> (DF)
19:49:03.073098 O 172.18.0.18.23 > 172.18.0.24.63306: R 0:0(0) ack 2423496500 win 0
```

The firewall sends a TCP RESET back to indicate that the port is not open. On the client side, you see a "connection refused" to reflect this, instead of an "connection timed out" in case the traffic is not allowed and is getting dropped by the firewall.

NMAP was run from the laptop as well, against OFW01 and OFW02, covering all 65535 ports in UDP and TCP. No abnormalities detected; the ports that were supposed to be filtered/closed with this rule (read: not covered by any previous rule) showed up as such. Telnet attempts to other ports on either firewall timed out as well, similarly to the telnet attempt to port 23 of the firewall.

3.2.14. Rule 20 and 21

These rules should enable everybody from the Internet to query the service IPs of the external DNS and WWW servers. GIAC networks (public/external and internal/corporate) are not able to query the service IPs per this rule.

While the DNS rule will be addressed in rule 24 to allow access from the public service network to the DNS service IP, the WWW01-SIP-IP is not mentioned later on. This should be addressed, so that employees from the corporate network are able to view the web server content the same way the customers are able to.

From the "Internet" (laptop plugged into VLAN 22 in front of the outside firewall), the standard tests were done and did not uncover any abnormalities: telnet to port 80 of the web server service IP, domain queries to UDP port 53 of the primary DNS server, all without problems. One of the tools used to test is, is DIG.

sample output:

```
bash# dig @ns.ekookie.com www.ekookie.com a
[...]
;; got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 6
;; flags: qr rd ra; QUERY: 1, ANSWER: 4, AUTHORITY: 4, ADDITIONAL: 2
;; QUERY SECTION:
;;      www.ekookie.com, type = A, class = IN

;; ANSWER SECTION:
 [...]
www.ekookie.com.            17m44s IN A     1.1.1.149
```

The lookup worked.

3.2.15. Rule 22

22	DNS01-SIP-E0	* Any	UDP domain-udp	accept	Log	OFWCluster	* Any	- DNS queries coming from our DNS server

This rule allows the primary DNS server to query ANY destination for DNS queries. While this will not cause any issues, its primary purpose is to be able to resolve hosts/domains/etc from DNS servers in the Internet. This should be reflected in this rule; the destination should be changed to only allow queries against Internet servers, not towards our own servers. It should not be permitted to query our internal DNS server, as the internal DNS carries internal / inside DNS information that should not be known to the public / world.

However, looking at the policy for the inside firewall cluster, there is no mention about public/outside VLANs being able to query the inside DNS server; hence, domain lookups from the primary DNS against the internal DNS will not be successful anyway.

Regardless, the rule should be modified as suggested above. This conclusion will also be part of the Evaluation section at the end of this assignment.

Testing the rule did not reveal anything abnormal; DNS resolutions towards outside DNS servers worked as expected. Again, DIG was used to test this rule.

3.2.16. Rule 23

23	DNS01-SIP-E0	INET_NTP_Servers_ALL	ntp	accept	Log	OFWCluster	* Any	- NTP sync from Internet NTP

Per rulebase, these are the NTP servers in the INET_NTP_Servers_ALL-group:

```
ntp-1.ece.cmu.edu
sundial.columbia.edu
time.berkeley.netdot.net
```

(These stratum-2-servers were taken from [NTPSERVERLIST] during the build of the rulebase)
The application used to test this was ntpdate.
Command:

```
bash-2.05# ntpdate ntp-1.ece.cmu.edu
```

client side output:

```
28 Jan 22:40:04 ntpdate[25067]: adjust time server 128.2.136.71 offset 0.002696 sec
```

interface: traffic will be seen on eth-s1p1c0, the Internet connection, as well as on eth-s1p2c0, the public server service network.

firewall: tcpdump output:

```
bash# tcpdump -i eth-s1p1c0 port 123
22:40:04.709727 O 1.1.1.148.123 > 128.2.136.71.123:  v3 client strat 0 poll 4 prec -6
(DF)
22:40:04.735246 I 128.2.136.71.123 > 1.1.1.148.123:  v3 server strat 2 poll 4 prec -16
(DF)
22:40:04.735537 O 1.1.1.148.123 > 128.2.136.71.123:  v3 client strat 0 poll 4 prec -6
(DF)
22:40:04.762357 I 128.2.136.71.123 > 1.1.1.148.123:  v3 server strat 2 poll 4 prec -16
(DF)
22:40:04.762569 O 1.1.1.148.123 > 128.2.136.71.123:  v3 client strat 0 poll 4 prec -6
(DF)
22:40:04.789307 I 128.2.136.71.123 > 1.1.1.148.123:  v3 server strat 2 poll 4 prec -16
(DF)
22:40:04.789578 O 1.1.1.148.123 > 128.2.136.71.123:  v3 client strat 0 poll 4 prec -6
(DF)
22:40:04.815530 I 128.2.136.71.123 > 1.1.1.148.123:  v3 server strat 2 poll 4 prec -16
(DF)
```

The same tests were done with the other ntp servers in the above mentioned group.

Conclusion: works as expected.

3.2.17. Rule 24

It was determined that the public servers should ntp sync with the GIAC DNS server in the first place, rather than going into the Internet for every ntp query. While this makes sense, it also needs to be considered that the DNS server is a stratum-3-server, since that server itself already pulls from stratum-2-servers. This might increase skew in timing.

However, the skew is pretty much irrelevant, also according to the NTP server list page [NTPSERVERLIST]: "In most cases the accuracy of the NTP secondary (stratum 2) servers is only slightly degraded relative to the primary servers and, as a group, the secondary servers may be just as reliable". Since we are pulling from 3 servers, this should not be a problem then.

The DNS was tested by running DIG on one of the public servers, in this case MX01, the service IP of which is 1.1.1.151. We tried to resolve www.ekookie.com on the GIAC nameserver, that is responsible / authoritative primary DNS for the ekookie.com-domain.

interface: eth-s1p2c0, public service network interface

command:
on the MX-relay MX01: `bash# dig @1.1.1.148 www.ekookie.com`

client side output:

```
; <<>> DiG 8.3 <<>> @1.1.1.148 www.ekookie.com
; (1 server found)
;; res options: init recurs defnam dnsrch
;; got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 6
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 3, ADDITIONAL: 3
;; QUERY SECTION:
;;      www.ekookie.com, type = A, class = IN

;; ANSWER SECTION:
www.ekookie.com.            1D IN A         1.1.1.149

;; AUTHORITY SECTION:
ekookie.com.                1D IN NS        ns.ekookie.com.
[...]
;; ADDITIONAL SECTION:
ns.ekookie.com.             1D IN A         1.1.1.148
[...]
```

tcpdump output on the firewall:

```
bash# tcpdump -ni eth-s1p2c0 host 1.1.1.148
tcpdump: listening on tun0
23:08:57.920782 1.1.1.151.1424 > 1.1.1.148.53:  6+ A? www.ekookie.com. (29)
23:08:57.932154 1.1.1.148.53 > 1.1.1.151.1424:  6* 1/3/3 A 1.1.1.149 (154) (DF)
```

Conclusion: this rule works as expected.

3.2.18. Rule 25

The purpose of this rule is to allow zone transfers and ntp-syncing, initiated by the internal DNS server towards the public primary DNS. In order to do a zone transfer, the secondary/internal DNS has to be able to connect to TCP (!) port 53 of the primary. Also, the secondary needs to be allowed to pull the zones from the primary by configuring the primary respectively and allowing the secondary to do this.

To test this, we will run DIG on the INSIDEDNS01 and will attempt to transfer the ekookie.com zone from DNS01 to it.

Interface:
eth-s1p4c0, the inside connection between the OFWCluster and the internal network

command:
on INSIDEDNS01:

```
bash# dig @1.1.1.148 ekookie.com axfr
```

client side output:

```
; <<>> DiG 8.3 <<>> @1.1.1.148 ekookie.com axfr
; (1 server found)
$ORIGIN ekookie.com.
@                       1D IN SOA       @ hostmaster.ekookie.com. (
                                        2003022001      ; serial
                                        8H              ; refresh
                                        2H              ; retry
                                        1W              ; expiry
                                        1D )            ; minimum

                        1D IN NS        DNS01
www01                   1D IN A         1.1.1.149
[...]
```

tcpdump output on the firewall:

```
bash# tcpdump -i eth-s1p4c0 host 1.1.1.148 and host 10.0.0.5
23:23:16.084193 O 10.0.0.5.41893 > 1.1.1.148.53: S 4162569956:4162569956(0) win 24820
<nop,nop,sackOK,mss 1460>
23:23:16.086007 O 10.0.0.5.41893 > 1.1.1.148.53: . ack 984082982 win 24820
23:23:16.086054 O 10.0.0.5.41893 > 1.1.1.148.53: P 0:2(2) ack 1 win 24820
23:23:16.087314 O 10.0.0.5.41893 > 1.1.1.148.53: P 2:31(29) ack 1 win 24820
23:23:16.097217 O 10.0.0.5.41893 > 1.1.1.148.53: . ack 1461 win 24820
23:23:16.097407 O 10.0.0.5.41893 > 1.1.1.148.53: . ack 2921 win 24820
23:23:16.099647 O 10.0.0.5.41893 > 1.1.1.148.53: . ack 5841 win 24820
23:23:16.100389 O 10.0.0.5.41893 > 1.1.1.148.53: . ack 10221 win 24820
[…]
```

Conclusion: this rule works as expected.

3.2.19. Rule 26

26	ISP_DNS-E0	DNS01-SIP-E0	TCP domain-tcp	accept	Log	OFWCluster	* Any	- allow ISP DNS server to pull zone files from our primary public DNS

Since we have another secondary DNS running at our ISP, their DNS server needs to be able to query our DNS01 on TCP port 53 for zone transfers as well.

Interface: eth-s1p1c0 on the outside firewall, the public Internet connection.

We verified the functionality by contacting GIAC's ISP and asking them to test this functionality. They emailed us back the output, we tcpdump'ed at the same time they were testing it, results are the same as for the previous communication, INSIDEDNS01 to DNS01: it worked.

Conclusion: this rule worked as expected as well.

3.2.20. Rule 27

27	INSIDEDNS01-NAT	ISP_DNS-E0	UDP domain-udp	accept	Log	OFWCluster	* Any	- allow our internal DNS server (NATed) to query our ISP secondary in case primary is down

This rule is needed to warrant that in case of an outage of DNS01, the INSIDEDNS01 is still able to resolve GIAC's public DNS side of things in case INSIDEDNS01 would not have a complete

zone. It was found that this rule needs improvement: INSIDEDNS01 needs to be permitted to transfer zones from the ISP-DNS server in order to keep up operations if the DNS01 would be unreachable and the giac.com-zone etc would expire on INSIDEDNS01 in the meantime. As a result of this, "domain-tcp" needs to be added in the outside and inside firewall policy as well as the border router access-lists that need to allow this communication to happen (INSIDEDNS01 -> ISP_DNS-E0 with domain-tcp). Also, the ISP_DNS needs to have INSIDEDNS01 in his configuration in order to allow zone transfers that way.

Other then that, standard DNS queries (non-zone-transfers, udp-only) worked fine from the INSIDEDNS01, proper operation was confirmed.

Conclusion: this rule needs to be changed to allow zone transfers from the ISP DNS.

3.2.21. Rule 28

This rule is for proper operation of the public mail/MX relay of GIAC. MX01 is the official MX for ekookie.com/giac and listed as such in GIAC's DNS. Incoming mail from the Internet goes to that server. Outgoing mail from the corporate network is processed in the next rule.

To test this rule, we place the laptop again into VLAN 22, the "router-to-firewall VLAN", in front of the outside firewall cluster and assign the IP 1.1.1.135 to the laptop.

interface: eth-s1p1c0, the public/Internet connection

command:
on the laptop, issue

```
bash# telnet 1.1.1.151 25
```

client side output:

```
Trying 1.1.1.151...
Connected to MX01.ekookie.com.
Escape character is '^]'.
220 MX01.ekookie.com ESMTP Postfix
quit
telnet>
```

As can be seen, the version of sendmail running on the MX cannot be found out by simply connecting to the mailserver. This is an instance of "security through obscurity".

The tcpdump output is nothing abnormal and therefore not listed here for space constraints.

Conclusion: this rule works as expected.

3.2.22. Rule 29

As mentioned before, this rule handles the MX01 <-> INSIDEMX communication, the forwarding of mails either from the inside to the outside or vice versa.

interface: one of the interfaces is eth-s1p4c0, the connection to the inside corporate network.

We tested this functionality the same way as in the rule before, via telnetting to the service IPs between the mail servers, to port 25/smtp each time; also verifying and confirming that the inside mail server disguises its sendmail version the same way. Information on how to remove the banner can be found here: [SENDMAILBANNER]

Conclusion: this rule worked as expected.

3.2.23. Rule 30

Since the outside management and the public server management networks are on different interfaces of the outside firewall cluster, we need to test all devices in either network to see if syslogging works.

To test whether syslog is receiving and accepting connections, we will use the "`logger`" command to send a syslog message to the syslog daemon running on SYSLOG01. The daemon should add the entry to its messages file, with the respective server's management IP address

First, we need to verify that the syslog.conf of the server configuration contains the following line. If not, we need to add it and re-initialize the syslogger so it re-reads its configuration file:

```
vi /etc/syslog.conf
add: user.err<TAB>@172.18.1.4
:wq!
kill -HUP <syslogd pid>
```

to have log messages with alert/facility level of "user.err" sent to the syslog daemon at 172.18.1.4 (service- IP address of SYSLOG01)

Then we send the log message:

```
bash# logger -p "user.err" "test from DNS01"
```

On the SYSLOG01 server, we have been doing a tail –f /var/adm/messages before we tried to send the log message from the audit laptop to see whether the server received the log message or not

```
user@SYSLOG01> tail -f /var/adm/messages

Dec 02 12:50:04 [172.18.0.20.22] user: test from audit laptop
```

It worked.

In both cases, interface eth-s1p4c0 on the outside firewalls will see all the traffic going towards the internal syslog server, therefore we will be running the tcpdump on that interface.

output on firewall:

```
bash# tcpdump -ni eth-s1p4c0 port 514
00:47:15.080839 O 172.18.0.20.32775 > 172.18.1.5.514:  udp 76 (ttl 253, id 33600)
```

We repeated this test for all other servers and devices and did not encounter any problems.

Conclusion: this rule works as expected.

3.2.24. Rule 31

This rule allows the general outside/public server – to – SecurID communication to the securid service (UDP 5500) to the SecurID servers as well as outside cluster – to SecurID communication to authenticate VPN users per SecurID. The rule should be restricted to not allow the general "outside management networks" to send securid traffic, rather only actual (registered) servers.

To test this rule, we will execute the installed ACE/agent on each server as well as authenticate per VPN client via SecurID.

In both cases, interface eth-s1p4c0 on the outside firewalls will see all the traffic going towards the internal SecurID servers, therefore we will be running the tcpdump on that interface.

We use DNS01 in this example, the other servers and devices look similar.

command:
```
bash#/opt/ace/prog/sdshell
```
client side output:
```
ENTER PASSCODE: xxxxxx
Passcode accepted.
```

tcpdump output:
```
bash# tcpdump -i eth-s1p4c0 port 5500
01:12:52.208843 O 172.18.0.20.34181 > 172.18.1.21.5500:  udp 124 (DF)
01:12:52.209437 I 172.18.1.21.5500 > 172.18.0.20.34181:  udp 124 (DF)
01:12:52.211265 O 172.18.0.20.34181 > 172.18.1.21.5500:  udp 508 (DF)
01:12:52.235650 I 172.18.1.21.5500 > 172.18.0.20.34181:  udp 508 (DF)
01:13:24.412661 O 172.18.0.20.34181 > 172.18.1.21.5500:  udp 508 (DF)
01:13:26.502355 I 172.18.1.21.5500 > 172.18.0.20.34181:  udp 508 (DF)
```

It works. This was also confirmed for communication between all the servers and devices and the SecurID replica server.

Conclusion: it works as expected.
Recommendation: Outside_Management_Networks should be replaced with the actual servers. This will be summarized at the end of this assignment.

3.2.25. Rule 32

The network devices (switches, router) cannot talk SecurID to the SecurID servers; they can use TACACSplus (49/tcp) to communicate with the SecurID servers.
Interface: the interface on the outside firewalls dedicated for outside devices is eth2c0, that is the interface we will be watching.

command:
from an allowed client, we connect to one of the switches per ssh
`bash# ssh -1 sol@172.18.0.4` // we need to force protocol version 1, that's the only version the Cisco switches support
client output:

```
sol@OSW01's password: XXXXXXXXXX
Password accepted.
OSW01>
```

tcpdump on the firewall:

```
bash# tcpdump -i eth2c0 port 49
01:19:47.988965 O 172.18.0.4.11345 > 172.18.1.21.49: S 201644436:201644436(0) win 4128
<mss 536>
01:19:47.989497 I 172.18.1.21.49 > 172.18.0.4.11345: S 1771408138:1771408138(0) ack
201644437 win 24656 <mss 1460> (DF)
01:19:47.989954 O 172.18.0.4.11345 > 172.18.1.21.49: . ack 1 win 4128
01:19:48.087300 O 172.18.0.4.11345 > 172.18.1.21.49: P 1:38(37) ack 1 win 4128
01:19:48.087508 I 172.18.1.21.49 > 172.18.0.4.11345: . ack 38 win 24656 (DF)
```

One can see how the switch (OSW01 / 172.18.0.4) establishes a connection to SECURID01's (172.18.1.21) TACACSplus service on port 49 and exchanges data.

This was also tested for the other network devices and was successful.

Conclusion: This rule works as expected.

3.2.26. Rule 33

This rule is for backup purposes. All servers and the TAPE server have to be able to communicate on the port ranges 13720-13750/tcp for Veritas NetBackup. According to the Backup admin, this is the minimal amount of ports open.

Interface: one of them is eth-s1p4c0, the connection to the corporate network.

command:
we will replace each server in the public management network with the audit laptop, one by one, as well as the TAPE server, to test if each server is able to reach the TAPE01-service IP on all the ports and vice versa. To test this, we will a) run NMAP over all TCP and UDP ports (1-65535 each) with connect-scans to establish full connectivity (and to verify that all those ports show up as open) and b) telnet manually to each port.
`bash# nmap -sT -p 1-65535 172.18.1.37` and `nmap -sU -p 1-65535 172.18.1.37` from each server IP in the public management network and nmaps from the TAPE-mgmt IP to the public management network range: `nmap -sT -p 1-65535 172.18.0.16/28` and `nmap -sU -p 1-65535 172.18.0.16/28`.

Sample output from nmap'ing TAPE01-sip from the DNS-IP:

```
Interesting ports on  (172.18.1.37):
(The 65505 ports scanned but not shown below are in state: filtered)
Port       State       Service
13720/tcp  open        unknown
13721/tcp  open        unknown
13722/tcp  open        unknown
```

[...]

The TCPdump is nothing out of the ordinary and therefore not shown here. The connect scans, telnet connection attempts, tcpdumps all indicate normal operation.

Conclusion: the rule works as expected.

3.2.27. Rule 34

interface: again, one of the interfaces on the outside firewall cluster dealing with this traffic will be eth-s1p4c0.

To test this, we initiate a telnet session from the WWW01 server to the service IP of the DB01-server. Connection gets established, TCPdumps are ok (not shown because of space constraints).

Conclusion: the rule works as expected.

3.2.28. Rule 35

To enable automated ssh, hostkeys need to be used (RSA hostkeys, since we are forcing SSH protocol version 2). During the audit we found that the shell account on the FTP server that accepted the hostkey without SecurID/password authentication was an actual user account with wheel-group privileges. This is a serious risk and therefore should be corrected. A measure to undertake would be to use a non-privileged shell account with minimal permissions for this procedure.

The rule was tested by telnetting to port 22/tcp from the DB server to the FTP server. Connection was established successfully, TCPdumps are nothing out of the ordinary and hence, not shown.

One of the interfaces used here is eth-s1p2c0, the public server service interface.

3.2.29. Rule 36

This rule allows the whole corporate network to access all public production servers per ssh and thus, should be restricted. The "corporate-10-network" also includes internal servers like the DNS and INSIDEMX, it is highly doubtful that one would need to ssh from e.g. the DNS server to the WWW server etc.

To test the rule, we telnet'ed from some corporate network machines to port 22/ssh of each public server's management IP, successfully. TCPDumps were nothing out of the ordinary.

Conclusion: this rule works as expected.

3.2.30. Rule 37

3.2.30.1. Conventions

In this audit, we had to determine how to scan/where/what to scan for what destination services available. Destination sites/ destination ports were selected as following:

www.sans.org, port 80/tcp
www.checkpoint.com, port 80/tcp
www.cisco.com, port 80/tcp
login.oscar.aol.com, port 5190/tcp (AOL server)
giactc.giac.org, port 443/tcp (https)

We will be reading the destination web servers from an input file, called http.txt.

The command line for port 80 as destination, don't ping:

```
root@scanner# nmap -sS -P0 -p 80 -T 3 -iL "http.txt"
```

The command line for port 5190 as destination, don't ping:

```
root@scanner# nmap -sS -P0 -p 5190 -T 3 login.oscar.aol.com
```

The command line for port 443 as destination, don't ping:

```
root@scanner# nmap -sS -P0 -p 443 -T 3 giactc.giac.org
```

3.2.30.2. the Audit

```
bash# nmap -sS -P0 -p 80 -T 3 -iL "http.txt"
bash# nmap -sS -P0 -p 5190 -T 3 login.oscar.aol.com
bash# nmap -sS -P0 -p 443 -T 3 giactc.giac.org

Reading target specifications from FILE: http.txt
Interesting ports on www.cisco.com (198.133.219.25):
Port       State       Service
80/tcp     open        http

Interesting ports on 63-100-47-46.secsup.org (63.100.47.46):
Port       State       Service
80/tcp     open        http

Interesting ports on 63-100-47-54.secsup.org (63.100.47.54):
Port       State       Service
80/tcp     open        http

Interesting ports on  (216.200.241.66):
Port       State       Service
80/tcp     open        http

Interesting ports on bucp1-vip-m.blue.aol.com (64.12.161.153):
Port       State       Service
```

```
5190/tcp   open        aol

Interesting ports on 63-100-47-54.secsup.org (63.100.47.54):
Port       State       Service
443/tcp    open        https

bash# telnet www.sans.org 80
Trying 63.100.47.46...
Connected to www.sans.org.
Escape character is '^]'.
GET /index.html
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<HTML><HEAD>
<TITLE>301 Moved Permanently</TITLE>
</HEAD><BODY>

bash# telnet www.checkpoint.com 80
Trying 216.200.241.66...
Connected to www.checkpoint.com.
Escape character is '^]'.
GET /index.html
<html>
<head>
<title>Check Point Software Technologies: We Secure the Internet</title>

bash# telnet www.cisco.com 80
Trying 198.133.219.25...
Connected to www.cisco.com.
Escape character is '^]'.
GET /index.htm
<!-- $Revision: 1.30 $ --><!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
<title>Cisco Systems, Inc</title>

bash# telnet login.oscar.aol.com 5190
Trying 64.12.161.185...
Connected to login.oscar.aol.com.
Escape character is '^]'.
*~ÜBLA
Connection closed by foreign host.

bash# telnet giactc.giac.org 443
Trying 63.100.47.54...
Connected to giactc.giac.org.
Escape character is '^]'.
```

As we expected here as well, randomly picked destinations with the services we intended to allow worked out fine. All the servers that were targeted responded at their picked destination port.

Conclusion: this rule works as expected.
Recommendation: it should be considered to disallow AOL-type connections between internal networks and the Internet.

3.2.31. Rule 38

The clean up rule. Similarly to rule 19, the stealth rule, all traffic not matching any of the previous rules (this time to any other destination) will get logged and dropped on this rule.

For example, a connection attempt to port 23/telnet-tcp of the webserver would be passed on to this rule.

3.3. Evaluation

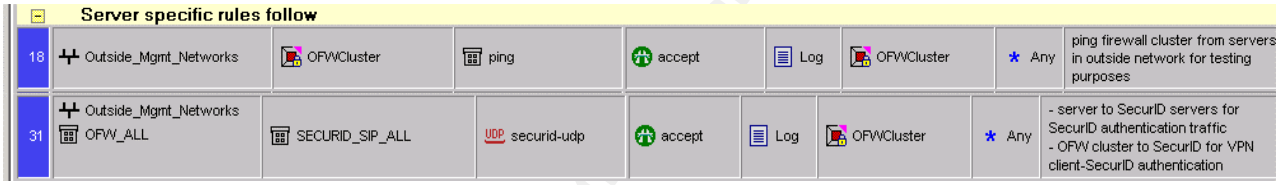
- Evaluate the audit. Based on your assessment (and referring to data from your assessment):
  - Provide an analysis of the audit results.
  - Make recommendations for improvements or alternate architectures.
  - Supportive diagrams are strongly recommended for this part of the assignment.

3.4. Analysis and Recommendations

3.4.1. Improvements / Corrections in the Firewall Policy

These are the "offending" rules":

Too generous in the source column (too many hosts allowed):



The rules should be restricted.

The improved rules could look like these:

Regarding rule 36:

This rule allows the whole corporate network to access all public production servers per ssh and thus, should be restricted. I suggest removing this rule and only let users access servers that use the VPN client.

Too generous in the destination column (too many hosts allowed)::

This rule allows the primary DNS server to query ANY destination for DNS queries. While this will not cause any issues, its primary purpose is to be able to resolve hosts/domains/etc from DNS servers in the Internet. This should be reflected in this rule; the destination should be changed to only allow queries against Internet servers, not towards our own servers.

The improved rule could look like this:

Not functioning properly:

INSIDEDNS01 needs to be permitted to transfer zones from the ISP-DNS server in order to keep up operations if the DNS01 would be unreachable and the giac.com-zone etc would expire on INSIDEDNS01 in the meantime. As a result of this, "domain-tcp" needs to be added in the outside and inside firewall policy as well as the border router access-lists that need to allow this communication to happen (INSIDEDNS01 -> ISP_DNS-E0 with domain-tcp). Also, the ISP_DNS needs to have INSIDEDNS01 in his configuration in order to allow zone transfers that way.

The corrected rule would look like this:

Insecure Service:

Recommendation: it should be considered to disallow AOL-type connections between internal networks and the Internet.

The improved rule would look like this:

It takes work and time to make these kind of rules work, but it should be invested in order to make the networks more secure. The basic strategy should always be "drop everything, only permit what is REALLY necessary".

Rule Order:

5	* Any	OFWCluster	IKE ESP FW1_topo	accept	Log	OFWCluster	* Any	- FW1_topo for topology downloads for vpn clients - IKE/ESP for IPSEC tunnels, client-to-gateway VPN
6	OFWCluster ResellerA_VPNGateway	OFWCluster ResellerA_VPNGateway	IPSEC	accept	Log	OFWCluster	* Any	IKE/ESP/AH for site-to-site VPN
7	OFWCluster ResellerB_VPNGateway	OFWCluster ResellerB_VPNGateway	IPSEC	accept	Log	OFWCluster	* Any	IKE/ESP/AH for site-to-site VPN
8	OFWCluster SupplierA_VPNGateway	OFWCluster SupplierA_VPNGateway	IPSEC	accept	Log	OFWCluster	* Any	IKE/ESP/AH for site-to-site VPN
9	OFWCluster SupplierB_VPNGateway	OFWCluster SupplierB_VPNGateway	IPSEC	accept	Log	OFWCluster	* Any	IKE/ESP/AH for site-to-site VPN
10	Reseller_Databases_ALL Supplier_Databases_ALL	FTP01-SIP-E0	ftp	Encrypt	Log	OFWCluster	* Any	- Resellers: download from FTP site - Suppliers: upload to FTP site all via VPN tunnels, encrypted.

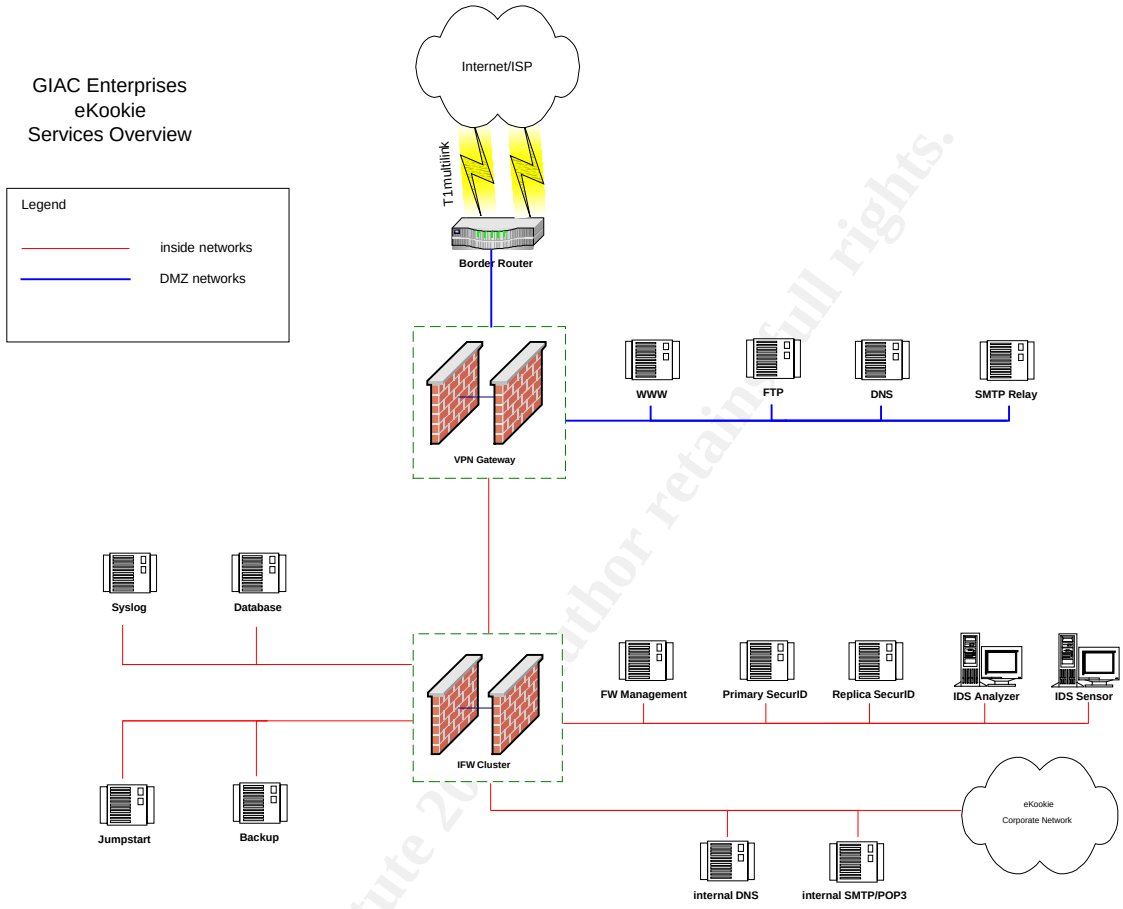
This works, but it may confuse the firewall admin if he/she has to do any log tracing for these rules. It may happen that traffic destined for rules 6-20 are getting logged at rule 5, that could be misleading if he/she tries to do any troubleshooting if there are issues. Also, "IKE" and "ESP" may be replaced with the"IPSEC" group object as that contains these services. The "IPSEC" group can be reduced to these two services, as the other ones (SKIP, AH, ISAKMP) are not used/supported/needed with our configuration of FW-1 NG.

For organizational reasons, a suggestion would be to change the order of these rules:

6	OFWCluster ResellerA_VPNGateway	OFWCluster ResellerA_VPNGateway	IPSEC	accept	Log	OFWCluster	* Any	IKE/ESP/AH for site-to-site VPN
7	OFWCluster ResellerB_VPNGateway	OFWCluster ResellerB_VPNGateway	IPSEC	accept	Log	OFWCluster	* Any	IKE/ESP/AH for site-to-site VPN
8	OFWCluster SupplierA_VPNGateway	OFWCluster SupplierA_VPNGateway	IPSEC	accept	Log	OFWCluster	* Any	IKE/ESP/AH for site-to-site VPN
9	OFWCluster SupplierB_VPNGateway	OFWCluster SupplierB_VPNGateway	IPSEC	accept	Log	OFWCluster	* Any	IKE/ESP/AH for site-to-site VPN
10	Reseller_Databases_ALL Supplier_Databases_ALL	FTP01-SIP-E0	ftp	Encrypt	Log	OFWCluster	* Any	- Resellers: download from FTP site - Suppliers: upload to FTP site all via VPN tunnels, encrypted.
5	* Any	OFWCluster	FW1_topo IPSEC	accept	Log	OFWCluster	* Any	- FW1_topo for topology downloads for vpn clients - IKE/ESP for IPSEC tunnels, client-to-gateway VPN

3.4.2. Improvements to the Network Architecture

Lets take another look at the network architecture:



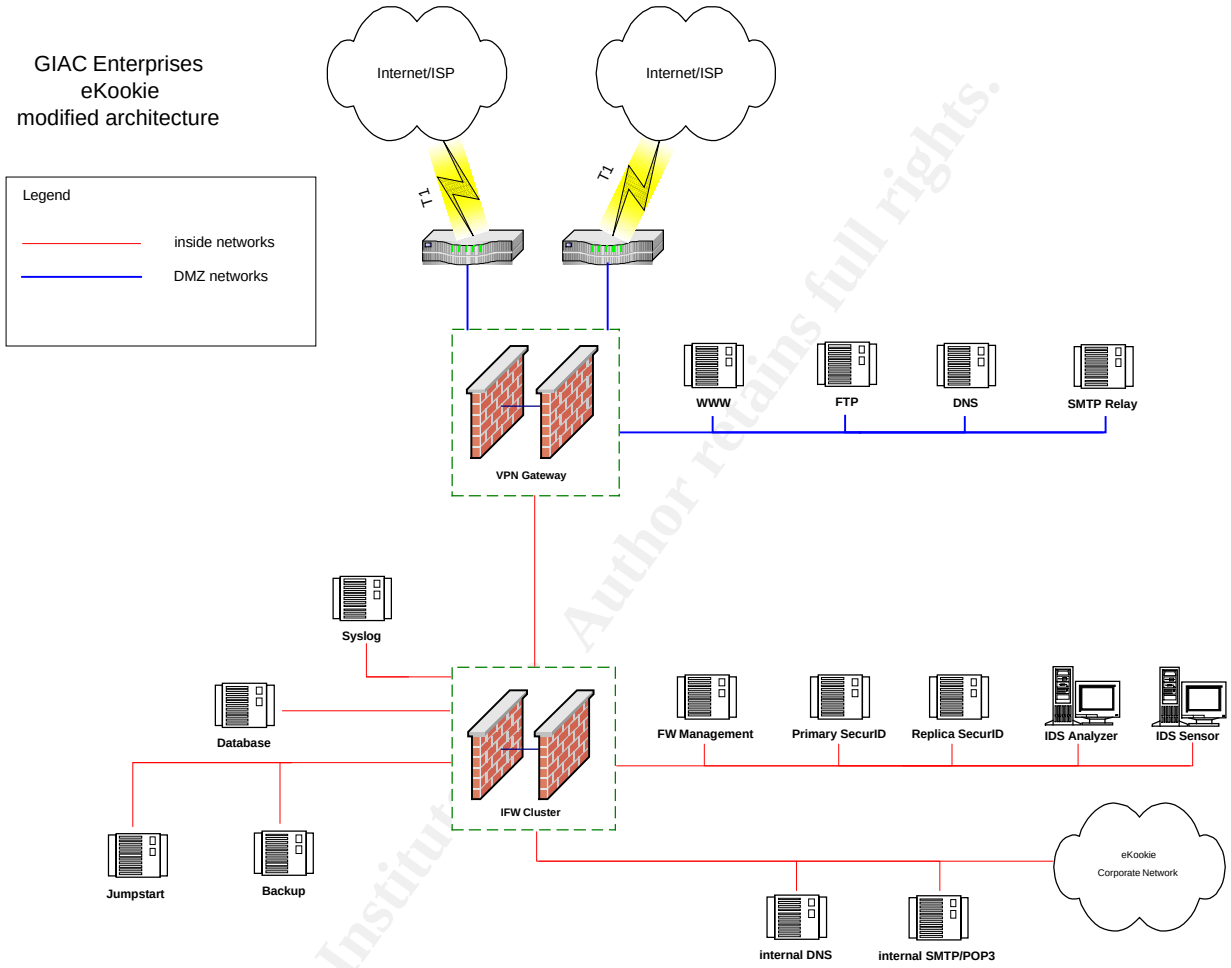
We would like to suggest to improve the network architecture in the following ways:

- add another switch to the outside and another one to the inside to form outside and inside switch clusters. This increases redundancy for the case one of the outside or inside switches should fail; the other one in the pair could take over.
- add another Cisco 3640 border router and make it a "router cluster" for redundancy, using HSRP for hot-standby routing (active/passive setup, passive takes over as soon as active one dies)
- split up the multilink and take one of the T1's and uplink to a second ISP. That way the Internet connection would be more reliable, if one ISP goes offline for any reason, the second ISP would still be available
- separate the syslog server from the database server and put them in separate VLANs/networks. That way, the log traffic coming across the service network doesn't interfere with production-database-to-webserver communication. Also, in case the database server would get compromised, hackers would not necessarily get access to the interfaces of the syslog server and may prevent them from taking over the syslog server and thus gaining even more information about the network by looking at the syslogs; also they might be able to disable the alerting system, remove traces from the central

syslog server etc. However, logs are still kept locally on the servers, so there may be still a way to trace hackers once the admins suspect the syslog server to be compromised

3.4.2.1. Alternative Network Architecture

The modified network architecture may look like this:



3.4.2.2. Cost considerations

The additional cost for modifying the network architecture would be calculated like this:

- additional router w/ T1 card
- additional switches for outside and inside, plus trunk connection (recommended: GBIC for gigabit trunk connection between switch pairs)
- additional ISP uplink, turn-up fees, monthly cost
- cost for contractors to set up the network changes, move the servers into different VLAN, modify routing etc; this would probably take up to two full work weeks (2 x 8 hours x 5 days x contractor-rate x number of contractors needed)

3.4.3. Other Suggestions

This should be an overview over what to do in order to keep the network secure.

- a Security Policy should be established so that the procedures for deploying new equipment / servers / rules are put in writing
- it is recommended to re-do the audit whenever new equipment is put in place, at least every six months
- all system administrators need to be subscribed to vendor and other security-related mailing lists in order to be up-to-date when new vulnerabilities are found and patches are released
- overall, it is recommended to create security administrator-like positions to centralize the issues that are security related (do audits, filter out important bugreports and patch notifications and forward them to the relevant people, watch the IDS and system logs)

4. Assignment 4 – Design Under Fire

Research and design the following three types of attacks against the architecture [you select]:

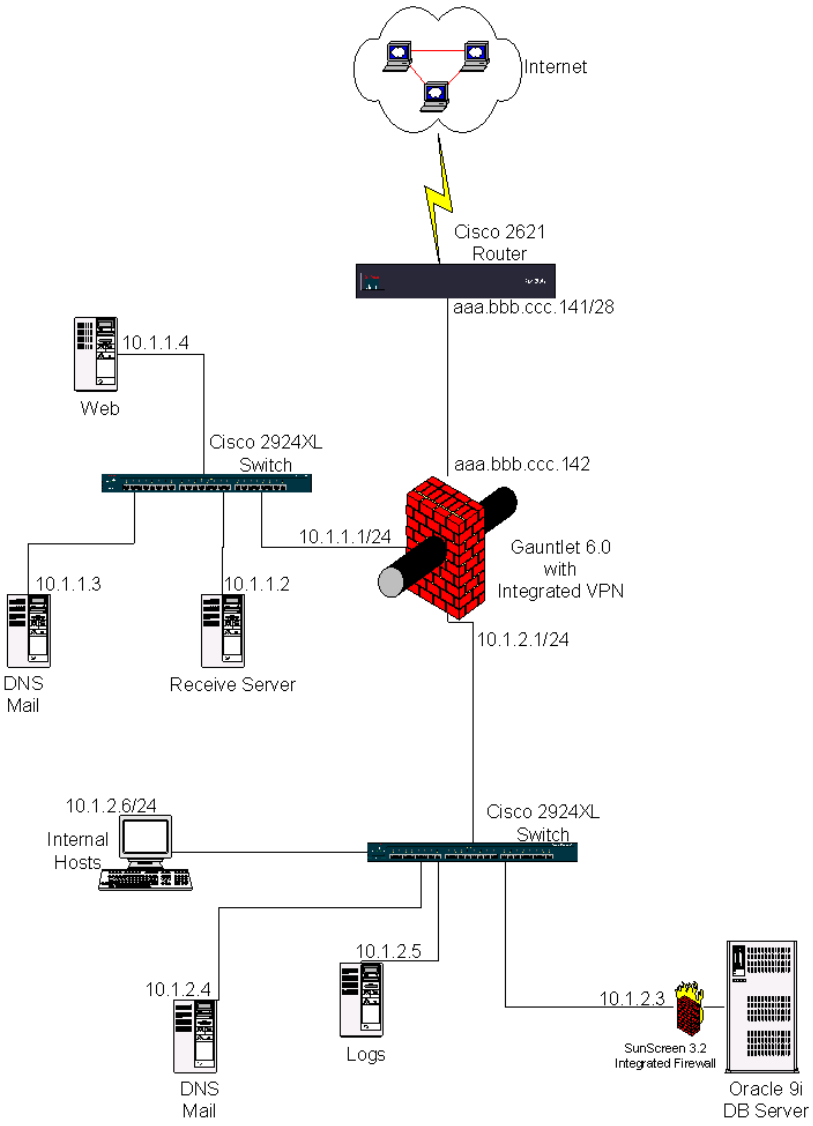
- An attack against the firewall itself.
  - Research and describe a vulnerability that has been found for the type of firewall chosen for the design.
  - Design an attack based on the vulnerability.
  - Explain the results of running that attack against the firewall.

- A denial of service attack.
  - Subject the design to an attack from 50 compromised cable modem/DSL systems.
  - Describe the countermeasures that can be put into place to mitigate the attack that you chose.

- An attack plan to compromise an internal system through the perimeter system.
  - Select a target and explain your reasons for choosing that target.
  - Describe the process to compromise the target.

4.1. Chosen Practical

http://www.giac.org/practical/Barry_Dowell_GCFW.doc,
Analyst #337 [DOWELL]



4.2. Attack against the Firewall itself

- An attack against the firewall itself.
  - Research and describe a vulnerability that has been found for the type of firewall chosen for the design.
  - Design an attack based on the vulnerability.
  - Explain the results of running that attack against the firewall.

There is a known vulnerability for the Gauntlet 6.0 Firewall.
[SECTRACK]
http://www.securitytracker.com/alerts/2001/Sep/1002321.html

[CERT]
http://www.cert.org/advisories/CA-2001-25.html

Gauntlet Firewall and PGP e-ppliances from Network Associates Have Buffer Overflows that Let Remote Users Get User-Level Operating System Access on the Firewall

Date: Sep 5 2001

Impact: Execution of arbitrary code via network, User access via network

Fix Available: Yes **Vendor Confirmed:** Yes

Version(s): Gauntlet for Unix versions 5.x, 6.0; PGP e-ppliance 300 series version 1.0, PGP e-ppliance 300 and 1000 series versions 1.5, 2.0

Description: Network Associates reported a vulnerability in their Gauntlet firewall. A buffer overflow allows a remote user to execute arbitrary code on the firewall and gain user-level access to the firewall's operating system.

A buffer overflow reportedly exists in the smap/smapd and CSMAP daemons. These daemons process SMTP-based e-mail transactions for both inbound and outbound e-mail. A remote user can trigger the buffer overflow and cause arbitrary shell commands to be executed on the firewall with the privileges of the daemon.

The security flaw apparently affects several Network Associates products.

The following products are reported to contain a vulnerability in the smap/smapd module:

Gauntlet for Unix versions 5.x
PGP e-ppliance 300 series version 1.0
McAfee e-ppliance 100 and 120 series

The following products are reported to contain a vulnerability in CSMAP:

Gauntlet for Unix version 6.0
PGP e-ppliance 300 series versions 1.5, 2.0
PGP e-ppliance 1000 series versions 1.5, 2.0
McAfee WebShield for Solaris v4.1

Impact: A remote user can gain user-level access to the operating system of the firewall.

[...]

As it turns out, the IT staff didn't get around to patching that firewall because of other "more urgent" projects that took away all their time.

[SECMAC]
http://www.infosecuritymag.com/2001/sep/digest10.shtml

"Involving two components of the firewall that handle e-mails, PGP said excessive amounts of data sent to the csmap SMTP proxy and smap/smapd can cause a system error. Properly crafted computer instructions appended to the text could give an attacker network access."

4.2.1. Designing the Attack

As mentioned above, the key is to send huge amounts of data to the SMTP proxy on the Gauntlet firewall and eventually, the daemon experiences a buffer overflow, elevating the privileges of the attacker to that of the daemon. If then the right kind of shellcode is sent, the attacker can execute commands on the firewall; he hacked it.

Having shell level access on this firewall as portrayed in the network design means to have access to most of the traffic / communication happening on the network – self-understood, a serious issue; it may be fatal for the business operations of the company.

To achieve this, we just have to send traffic to port 25 of the firewall, the smtp proxy, from anywhere in the Internet. Mail may be sent from anywhere in the Internet to employees of the company.

I was not able to find any code examples for this attack on the Internet, therefore the following pseudocode would illustrate how a possible exploit could be written.

4.2.1.1. Pseudocode for the Exploit

The basic principle of the code should be to take advantage of the vulnerability, that stated "A buffer overflow reportedly exists in the smap/smapd and CSMAP daemons. These daemons process SMTP-based e-mail transactions for both inbound and outbound e-mail. A remote user can trigger the buffer overflow and cause arbitrary shell commands to be executed on the firewall with the privileges of the daemon."

The attack consists of four phases:

1) establish a TCP connection to port 25 of the proxy
2) send enough arbitrary data to the port to overflow the buffer
3) send shellcode that, if 2) was successful, gets written into the memory of the proxy server and results in the kernel executing the shellcode, thus spawning a remote shell on the proxy for the hacker. The level of access on the proxy server will be the same the proxy daemon has (the attacker will have the same privileges as the proxy daemon)
4) execute commands on the proxy server

The syntax for the pseudo code is arbitrarily chosen and does not follow any specific standards (which is beyond the scope of this paper, it should merely illustrate the steps required from a 10,000 feet view).

Pseudo code for this attack may look like this (program is called "ExploitGauntletSMAP"):

comments are preceded with "//"

```
// "ExploitGauntletSMAP" - Pseudocode
// START MAIN
Print("ExploitGauntletSMAP v0.01 by EvilHacker")
TargetIP=ReadInputFromCommandLine(firstArg)
TargetOS=ReadInputFromCommandLine(secondArg)

If (TargetIP equal ""  or !IsValidIP(TargetIP) or \
      TargetOS notequal "" or !isValidOS(TargetOS))
// fail and display usage if IP and/or OS supplied are not valid
{
      DisplayUsage()
      Exit
}
Else
      Print("Got IP TargetIP, OS TargetOS.")

connectOK=ConnectToPort(TargetIP,25,"FullConnect")
If(!connectOK) // if the function - connect was not successful
      Die("Connect Failed.")
Else
      Print("Connect OK. Sending garbage.")

Do
      SendGarbageToPort() // send arbitrary data (like 0x0 or
                          // something) to fill up buffer of daemon
While(ConnectToPort(TargetIP,25,"TestOnly"))
Print("Done sending garbage. Sending shell code.")
SendOSSpecificShellcode(TargetOS)
Print("Done sending shell code. Sending command to test (id).")
Output=TestShell()
If(!Output) // if output does not contain anything
      Print("Failed :(")

// END MAIN


Function IsValidIP // Is this a valid IP?
{
      IP=ValueFromFunctionCall
      If(IP consists of /Digits\.Digits\.Digits\.Digits/)
            Print("IP ok.")
            Return 1
      Else
            Return 0
}
```

```
Function IsValidOS // Is this one of the supported OS?
{
      OS=ValueFromFunctionCall
      If(OS In {"Solaris","HP-UX")
            Print("OS ok.")
            Return 1
      Else
            Return 0
}

Function SendOSSpecificShellcode // pick shellcode
{
      OS=ValueFromFunctionCall
      Print("Got Shellcode.")
      Return ShellCodeForOS(OS)
}

Function ConnectToPort // connect to Port
{
      IP=ValueFromFunctionCall
      Port=ValueFromFunctionCall
      ModeOfConnect=ValueFromFunctionCall

      connectOK=Connect(IP,Port) //return 1 if OK, 0 if not OK

      If(ModeOfConnect equals "TestOnly")
            Disconnect(IP,Port) // to free up port

      return connectOK //contains either 1 - "connect ok" or
                       //0 -"connect failed"
}

Function TestShell // tests whether buffer overflow + shellcode worked
{
      Output=EchoToPort("id") // send Unix command and return output
      return Output // print output, results of the command execution
}

Function DisplayUsage // help
{
      Print << EOI
Usage: ./ExploitGauntletSMAP <DestinationIP> <OS>
Where IP is in the form 1.1.1.1
and OS is either "Solaris" or "HP-UX"

e.g. ./ExploitGauntletSMAP 127.0.0.1 "HP-UX"
EOI
      return
}
```

4.2.2. Execution of Attack

Assuming we would use a compiled version of above script, it may look like this:

```
evil@hacker# ./ExploitGauntletSMAP aaa.bbb.ccc.142 "Solaris"
ExploitGauntletSMAP v0.01 by EvilHacker
```

```
Got IP aaa.bbb.ccc.142, OS Solaris
Connect OK. Sending garbage.
Done sending garbage. Sending shell code.
Done sending shell code. Sending command to test (id)
Failed :(
evil@hacker#
evil@hacker#
```

4.2.3. Results of the Attack

Since the proxy has been patched with the latest updates available, the vulnerability does not exist anymore and thus, the script failed.

4.2.4. Countermeasures To Vulnerability

- the servers should always be on the latest possible patch level if there are any security vulnerabilities published, administrators need to be subscribed to vendor-relevant and general security mailing-lists etc to be able to keep up-to-date
- another possible countermeasure (to this vulnerability in specific) would be to disable the proxy service for the mail service, which would, however, result in an interruption in service

4.3. Denial of Service Attack

- A denial of service attack.
  - Subject the design to an attack from 50 compromised cable modem/DSL systems.
  - Describe the countermeasures that can be put into place to mitigate the attack that you chose.

4.3.1. Tribal Flood Network 2K

The attack will be done by making use of the TFN2K system, the "Tribe Flood Network 2000" client and server mechanism.

4.3.1.1. Description

[TFN]

```
Overview - What is TFN2K?

TFN2K allows masters to exploit the resources of a number of agents in
order to coordinate an attack against one or more designated targets.
Currently, UNIX, Solaris, and Windows NT platforms that are connected
to the Internet, directly or indirectly, are susceptible to this
attack. However, the tool could easily be ported to additional
platforms.

TFN2K is a two-component system: a command driven client on the master
and a daemon process operating on an agent. The master instructs its
agents to attack a list of designated targets. The agents respond by
```

flooding the targets with a barrage of packets. Multiple agents, coordinated by the master, can work in tandem during this attack to disrupt access to the target. Master-to-agent communications are encrypted, and may be intermixed with any number of decoy packets. Both master-to-agent communications and the attacks themselves can be sent via randomized TCP, UDP, and ICMP packets. Additionally, the master can falsify its IP address (spoof). These facts significantly complicate development of effective and efficient countermeasures for TFN2K.

TFN2K - The Facts

- Commands are sent from the master to the agent via TCP, UDP, ICMP, or all three at random.
  Targets may be attacked with a TCP/SYN, UDP, ICMP/PING, or BROADCAST PING (SMURF) packet flood. The daemon may also be instructed to randomly alternate between all four styles of attack.

- Packet headers between master and agent are randomized, with the exception of ICMP, which always uses a type code of ICMP_ECHOREPLY (ping response). Unlike its predecessors, the TFN2K daemon is completely silent; it does not acknowledge the commands it receives.

  Instead, the client issues each command 20 times, relying on probability that the daemon will receive at least one. The command packets may be interspersed with any number of decoy packets sent to random IP addresses.

- TFN2K commands are not string-based (as they are in TFN and Stacheldraht). Instead, commands are of the form "+<id>+<data>" where <id> is a single byte denoting a particular command and <data> represents the command's parameters. All commands are encrypted using a key-based CAST-256 algorithm (RFC 2612). The key is defined at compile time and is used as a password when running the TFN2K client.

- All encrypted data is Base 64 encoded before it is sent. This holds some significance, as the payload should be comprised entirely of ASCII printable characters. The TFN2K daemon uses this fact as a sanity-test when decrypting incoming packets.

- The daemon spawns a child for each attack against a target. The TFN2K daemon attempts to disguise itself by altering the contents of argv[0], hereby changing the process name on some platforms. The falsified process names are defined at compile time and may vary from one installation to the next. This allows TFN2K to masquerade as a normal process on the agent. Consequently, the daemon (and its children) may not be readily visible by simple inspection of the process list. All packets originating from either client or daemon can be (and are, by default) spoofed.

- The UDP packet length (as it appears in the UDP header) is three bytes longer than the actual length of the packet.

- The TCP header length (as it appears in the TCP header) is always zero. In legitimate TCP packets, this value should never be zero.

- The UDP and TCP checksums do not include the 12-byte pseudo-header, and are consequently incorrect in all TFN2K UDP and TCP packets.

4.3.1.2. Usage

The system consists of tfn, the client and td, the daemon. The daemon needs to be installed on helper/slave machine that were compromised earlier (see next section on this)

Running tfn without options yields the usage screen:

```
usage: ./tfn <options>
[-P protocol]   Protocol for server communication. Can be ICMP, UDP or TCP.
                Uses a random protocol as default
[-D n]          Send out n bogus requests for each real one to decoy targets
[-S host/ip]    Specify your source IP. Randomly spoofed by default, you need
                to use your real IP if you are behind spoof-filtering routers
[-f hostlist]   Filename containing a list of hosts with TFN servers to contact
[-h hostname]   To contact only a single host running a TFN server
[-i target string]      Contains options/targets separated by '@', see below
[-p port]               A TCP destination port can be specified for SYN floods
<-c command ID> 0 - Halt all current floods on server(s) immediately
                1 - Change IP antispoof-level (evade rfc2267 filtering)
                    usage: -i 0 (fully spoofed) to -i 3 (/24 host bytes spoofed)
                2 - Change Packet size, usage: -i <packet size in bytes>
                3 - Bind root shell to a port, usage: -i <remote port>
                4 - UDP flood, usage: -i victim@victim2@victim3@...
                5 - TCP/SYN flood, usage: -i victim@... [-p destination port]
                6 - ICMP/PING flood, usage: -i victim@...
                7 - ICMP/SMURF flood, usage: -i victim@broadcast@broadcast2@...
                8 - MIX flood (UDP/TCP/ICMP interchanged), usage: -i victim@...
                9 - TARGA3 flood (IP stack penetration), usage: -i victim@...
                10 - Blindly execute remote shell command, usage -i command
```

4.3.2. Compromising Machines that will Execute the Attack

In order to attack the destination network, we first need to compromise 50 hosts that will execute the actual tribe-flood-network attack. There are several criteria that should be fulfilled for a potential machine to be able to successfully leverage the attack against the destination network

- victim/slave machine should have lots of bandwidth so it can fill up a sufficient amount of bandwidth of the uplink of the target network
- should be easy to compromise so we don't spend too much time on compromising those victim/slave machines

We figure that preferred targets would be privately owned machines, not protected by corporate firewalls etc, hence the ideal victim would look like this:

- has a cable/dsl connection to the Internet
- runs an OS like Windows 9x, XP, ME or the like
- either runs easy to compromise servers (take IIS) or uses a browser like Internet Explorer. Either way, it is likely that we have some sort of tool / method that is relatively easy to implement so we can take over /

infiltrate that machine and deploy our TribalFloodNetwork client software.

In order to find the targets, we find out the cable/DSL IP network pools used by a couple of major Internet Service Providers and scan those with tools like nmap and nessus, then attack the "worthy" victims or otherwise compromise them and take over those machines / install our software on them.

Some methods of compromising those machines can be derived from security advisories like this one:

[JAVA security]
"[2] - Bytecode Verifier vulnerability
(it affects MS Internet Explorer 4.0-6.0 including VM build 3805)

Its successful exploitation allows for complete circumvention of the Java type safety rules. In a result of this, applet sandbox restrictions can be also escaped and malicious actions can be taken on the computer of the victim user."

We can be sure that we find at least 50 hosts on the Internet that do not have the latest patches / fixpacks for any number of vulnerabilities installed so it should not be a problem to find our 50 hosts that we will use against the target network.

4.3.3. The Attack

We know that the target firewall has the IP aaa.bbb.ccc.142. We have 50 compromised hosts we have the IPs from (that are stored in a text file we will feed to tfn on the command line with the –f switch). Not caring about which port should get TCP synflooded (we choose TCP synflooding and random port picking), our command to start the attack would look like this (including the output of the client):

```
evil@hacker# ./tfn -f hostlist.txt -c 5 -i aaa.bbb.ccc.142

        Protocol      : random
        Source IP     : random
        Client input  : list
        Target(s)     : aaa.bbb.ccc.142
        Command       : commence syn flood, port: random

Sending out packets: .
evil@hacker# |
```

This will result in the firewall get DoSed from all the compromised agent hosts.
The One-ISP-Uplink and the Cisco 2621 border router will not be able to handle this attack and get flooded. At this point it does not even matter anymore whether the firewall can handle it or not (maybe it can, maybe it cannot), since the attacker has achieved his goal: essentially bringing the network offline since the Internet connection is at capacity.

4.3.4. Countermeasures To DDoS Attack

- Immediately contact ISP upon determination that a DDOS attack is happening and the network is significantly degraded. Work with ISP to find out where the attacks are coming from and try to thwart off the zombies/attacking machines at the ISP level. Eventually, have the ISP contact its uplink provider (if applicable) and make them work together. It may be that the IPs are spoofed so more in-depth research needs to be done (MAC address backtracking etc). This could prove to be a very difficult/complex task.
- Also it could be tried to setup CAR (Commited Access Rate)-Limiting on the ISP uplink, essentially limiting the amount of traffic that can be sent for certain protocols (e.g limiting ICMP traffic, TCP traffic on a per-protocol basis). Source: [CAR]
- configure the border router to use TCP Intercept [CISCO-INTERCEPT], if it is a Cisco router. TCP Intercept can be either used in intercept or watch-mode. The intercept mode puts the router in the line of communication flow, as it takes every incoming TCP SYN request, validates it, then connects it with the intended destination. The watch mode monitors incoming SYN requests passively and takes action (i.e. terminates the connection) if the connection request does not get handled in a timely manner
- a second ISP uplink would be helpful in moving the issue of the DoS attack away from the border router, more towards the core firewall which would be able to handle more traffic than the single ISP uplink itself. A border router pair, using HSRP as protocol for intercommunication, combined with another ISP uplink would somewhat remedy the situation, albeit not fix it
- another thing to consider would be to add a backup Internet connection, like a shared PVC/56k standby connection that would get used in such a case. Not enough to do serious business, but at least there would be some connectivity
- More info can be found at [FIRSTAID], proposed security measures against DDOS attacks, written by Mixter, the author of Targa, TFN and TFN2k

4.4. Compromise an Internal System

- An attack plan to compromise an internal system through the perimeter system.
  - Select a target and explain your reasons for choosing that target.
  - Describe the process to compromise the target.

4.4.1. Target Selection

I have decided to select the web server in the GIAC network. It is running Apache with the latest version, however, it is not mentioned what platform the web server is running on, or what version of SSL the webserver is using. Therefore, we assume the web server is running Apache on an Intel-based Linux box with OpenSSL < 0.9.6e with SSL v2 enabled. Given this, a vulnerability affecting Apache/modssl is applicable: the Apache/mod_ssl worm (bugtraq/slapper/apache-worm).
[CERT-APACHE]

```
CERT® Advisory CA-2002-27 Apache/mod_ssl Worm
Original release date: September 14, 2002
Last revised: October 11, 2002
Source: CERT/CC

[…]
```

```
Overview
The CERT/CC has received reports of self-propagating malicious code
which exploits a vulnerability (VU#102795) in OpenSSL. This malicious
code has been referred to as Apache/mod_ssl worm, linux.slapper.worm
and bugtraq.c worm. Reports received by the CERT/CC indicate that the
Apache/mod_ssl worm has already infected thousands of systems. There
are currently at least three known variants of this worm in
circulation.

I. Description
The Apache/mod_ssl worm is self-propagating malicious code that
exploits the OpenSSL vulnerability described in VU#102795. This
vulnerability was the among the topics discussed in CA-2002-23 Multiple
Vulnerabilities In OpenSSL. While this OpenSSL server vulnerability
exists on a wide variety of platforms, the Apache/mod_ssl worm appears
to work only on Linux systems running Apache with the OpenSSL module
(mod_ssl) on Intel architectures.

The Apache/mod_ssl worm scans for potentially vulnerable systems on
80/tcp using an invalid HTTP GET request. When a potentially vulnerable
Apache system is detected, the worm attempts to connect to the SSL
service via 443/tcp in order to deliver the exploit code. If
successful, a copy of the malicious source code is then placed on the
victim server, where the attacking system tries to compile and run it.
Once infected, the victim server begins scanning for additional hosts
to continue the worm's propagation.
[…]

II. Impact
Compromise by the Apache/mod_ssl worm indicates that a remote attacker
can execute arbitrary code as the apache user on the victim system. It
may be possible for an attacker to subsequently leverage a local
privilege escalation exploit in order to gain root access to the victim
system.
```

In modification of the original worm, we will introduce these changes:

- instead of the peer-to-peer functionality, we will spawn a shell that replaces the web server running on port 80. The reason for this is that the gauntlet firewall only allows traffic to port 80 and 443, so we need to pick one of those two in order to be able to communicate with the remote shell
- the shell will be a precompiled static binary suitable for Linux/x86, since we cannot assume that a compiler is installed on the target machine (should not be if hardened properly).

We found a source that is customized for Apache on Linux x86, it is called "apache-ssl-bug.c" and can be found at packetstormsecurity.com. [WORM-SOURCE]

Analysing the source, the program follows these steps:

Phase I: reconnaissance
1) target selection, get destination IP and destination OS from the command line (or try to find the target OS by analysing the strings it gets back when trying to connect in 2)
2) connect to port 443 of the destination IP (by default)
3) send the following string: GET / HTTP/1.1\r\n\r\n
   This results in an error message from the webserver, with which the target OS can be determined.

Phase II: buffer overflow the server
4) send an SSL handshake to the destination, SSLv2
5) the buffer overflow happens in this phase (v2 handshake, per [CERT-MODSSL]
6) here we place our alteration of the source. Instead of transmitting the shell-source to the server, we transmit the precompiled shell in uuencoded form and store it in the /tmp/ directory of the server
7) because of the buffer overflow, we now have shell level access with the privileges of the apache web server. This enables us to:
8) uudecode the binary
9) kill the webserver running on port 80
10) launch the shell, binding it to port 80 (that's the way we have set it in the source of the shell prior to compilation

That is it. Now we have a remote shell running on port 80 of the web server.

4.4.2. Executing the attack

This stage consists of the following phases:

1) prepare the shell source, precompile it, uuencode it
2) place the uuencoded version in the source of the apache-worm source in hex form. This way it can be insert into the packets we transmit to the destination
3) execute the attack / launch the program
4) wait for it to finish, then try to connect to port 80 of the destination and see if a shell has been launched.

A successful attack would look like this (output taken from apache-ssl-bug source code and amended appropriately:

```
evil@hacker# ./apache-ssl-bug -v aaa.bbb.ccc.142

Apache & OpenSSL 0.9.6 Exploit
Made by andy^ after the bugtraq.c worm

Trying to exploit aaa.bbb.ccc.142
DONE
evil@hacker#
```

After this is finished, we connect to port 80 of the destination webserver:

```
evil@hacker# telnet aaa.bbb.ccc.142 80
Trying aaa.bbb.ccc.142...
Connected to aaa.bbb.ccc.142.
```

```
Escape character is '^]'.
$
```

Done, in theory. In reality, the attack will fail because it seems the destination is patched with the latest modssl. So the actual program output will look like this:

```
evil@hacker# ./apache-ssl-bug -v aaa.bbb.ccc.142

Apache & OpenSSL 0.9.6 Exploit
Made by andy^ after the bugtraq.c worm

Trying to exploit aaa.bbb.ccc.142
Could not connect
FAILED
evil@hacker#
```

4.4.3. Countermeasures

- always have the systems on the latest patchlevels
- perform security audits in regular intervals
- for this attack in specific (if the system would not be able to get upgraded to the latest modssl version): disable SSLv2
  Disabling SSLv2 handshaking will prevent exploitation of VU#102795, the openssl bufferflow vulnerability [CERT-MODSSL]
- also it is a must for the system administrators to be subscribed to the appropriate security mailing lists (e.g. bugtraq and vendor specific mailing lists) in order to be up to date on the latest patches that are available

4.4.4. Will we be noticed?

- there is no Intrusion Detection System installed, per Barry's documentation. So the alerting would be delayed
- however, customers and employees would most likely realize very quickly that the webservice is not available anymore and would start troubleshooting
- also, as a prelude to the attack, error-log entries like the following may be seen on the web server, in the httpd-error-log:

```
[Fri Jan 1 18:33:31 2003] [error] mod_ssl: SSL handshake failed: HTTP
spoken on HTTPS port; trying to send HTML error page (OpenSSL library
error follows)
[Fri Jan 1 18:33:31 2003] [error] OpenSSL: error:1407609C:SSL
routines:SSL23_GET_CLIENT_HELLO:http request [Hint: speaking HTTP to
HTTPS port!?]
[Fri Jan 1 18:33:34 2003] [error] [client x.x.x.x] request failed:
error reading the headers
```

This is the typical signature of this worm. Of course, the HTTP request itself may be changed, it would not matter as long as the actual request generates an error response of the webserver that could be used to find out the version/OS.

Since the network has a syslog facility, it would be likely that this message would get forwarded to the syslog server (if the webserver syslog daemon is configured to do so and the syslog server

acts upon messages like this) and staff might be able to see this. Oftentimes, however, standard HTTP server error messages get disregarded so the attack couldn't be avoided, but this may be a piece of the puzzle for later on, when the actual incident has to be handled / finding out what happened has to occur.

5. References

[FW1]
http://www.checkpoint.com/products/protect/firewall-1.html
- Checkpoint Firewall-1 home page

[CISCO]
http://www.cisco.com
- Cisco home page

[INREACH]
http://www.itouchcom.com/products/index.cfm?cat=ts
- InReach Itouch Terminal Server home page

[YASSP]
http://www.yassp.org/
- "Yet Another Solaris Security package" home page

[SUN]
http://www.sun.com
- Sun Microsystems home page

[RSA]
http://www.rsasecurity.com/products/securid/
- SecurID Product Information

[SMARTDEFENSE]
http://www.checkpoint.com/products/downloads/smartdefense_datasheet.pdf
- SmartDefense Product Information

[VRRP]
http://www.ietf.org/html.charters/vrrp-charter.html
- VRRP Charter

[MULTIPATHING]
http://www.netsys.com/library/papers/Multipathing-updt1.pdf
- Multipathing Whitepaper

[RFC1918]
http://www.isi.edu/in-notes/rfc1918.txt
- RFC describing reserved networks

[IANA]
http://www.iana.org/assignments/ipv4-address-space
- IANA reserved IP address networks

[MULTILINK]
http://www.cisco.com/en/US/partner/products/hw/modules/ps2033/products_white_paper09186a0080091d4b.shtml
- Alternatives for High Bandwidth Connections Using Parallel T1/E1 Links

[CISCO1]
http://rr.sans.org/firewall/blocking_cisco.php
- Securing the Perimeter with Cisco IOS 12 Routers

[CISCO2]
http://pasadena.net/cisco/secure.html
- sample router configuration

[CISCO3]
http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/12cgcr/rbkixol.htm
- complete Cisco IOS 12.x command reference guide

[NMAP]
http://www.insecure.org/nmap/data/nmap_manpage.html
- NMAP documentation

[BANNER]
http://www.kerr1.com/docs/firewall/router2.htm
- Cisco Router Hardening Step-by-Step

[NSAC]
http://nsa1.www.conxion.com/cisco/guides/cis-2.pdf
- NSA Cisco Router Configuration Guide

http://www.sans.org/top20.htm
- The Twenty Most Critical Internet Security Vulnerabilities, SANS

[NTPSERVERLIST]
http://www.eecis.udel.edu/~mills/ntp/clock2a.html
- list of public accessible NTP servers

[SENDMAILBANNER]
http://secinf.net/unix_security/Securing_Solaris.html
- Securing Solaris

[HPING]
http://www.hping.org/
- HPING homepage

[SECTRACK]
http://www.securitytracker.com/alerts/2001/Sep/1002321.html
- "Gauntlet Firewall and PGP e-ppliances from Network Associates Have Buffer Overflows"

[CERT]
http://www.cert.org/advisories/CA-2001-25.html
- CERT® Advisory CA-2001-25 Buffer Overflow in Gauntlet Firewall allows intruders to execute arbitrary code

[RPC]
http://www.cert.org/incident_notes/IN-99-04.html
- Similar Attacks Using Various RPC Services

[TFN]
http://www.packetstormsecurity.org/distributed/TFN2k_Analysis-1.3.txt
- TFN2K analysis by AXENT

[SECMAC]
http://www.infosecuritymag.com/2001/sep/digest10.shtml#news1
- Article about "Mandatory" Patch Released for Gauntlet Firewall

[DOWELL]
http://www.giac.org/practical/Barry_Dowell_GCFW.doc
- Barry Dowell's Practical

[JAVA security]
http://msgs.securepoint.com/cgi-bin/get/bugtraq0211/255.html
- [LSD] Java and JVM security vulnerabilities advisory

[CERT-ORACLE]
http://www.cert.org/advisories/CA-2002-08.html
- CERT® Advisory CA-2002-08 Multiple Vulnerabilities in Oracle Servers

[FRIENDGREETING]
http://securityresponse.symantec.com/avcenter/venc/data/friendgreetings.html
- W32.Friendgreet.worm

[FIRSTAID]
http://www.defcon.tv/distributed/firstaid.txt
- Mixter's 10 proposed security measures against DDOS attacks

[CISCO-INTERCEPT]
http://www.cisco.com/en/US/partner/products/sw/iosswrel/ps1831/products_configuration_guide_chapter09186a00800d9818.html
- Configuring TCP Intercept (example for IOS 12.1)

[CAR]
http://www.cisco.com/en/US/partner/tech/tk543/tk545/tk764/tech_protocol_home.html
- CAR Rate Limiting, Cisco

[CERT-APACHE]
http://www.cert.org/advisories/CA-2002-27.html
- CERT® Advisory CA-2002-27 Apache/mod_ssl Worm

[CERT-MODSSL]
http://www.kb.cert.org/vuls/id/102795
- OpenSSL servers contain buffer overflow during SSL2 handshake process

[WORM-SOURCE]
http://packetstormsecurity.org/0209-exploits/apache-ssl-bug.c
- source for Apache/modssl-exploit / worm

EOF.