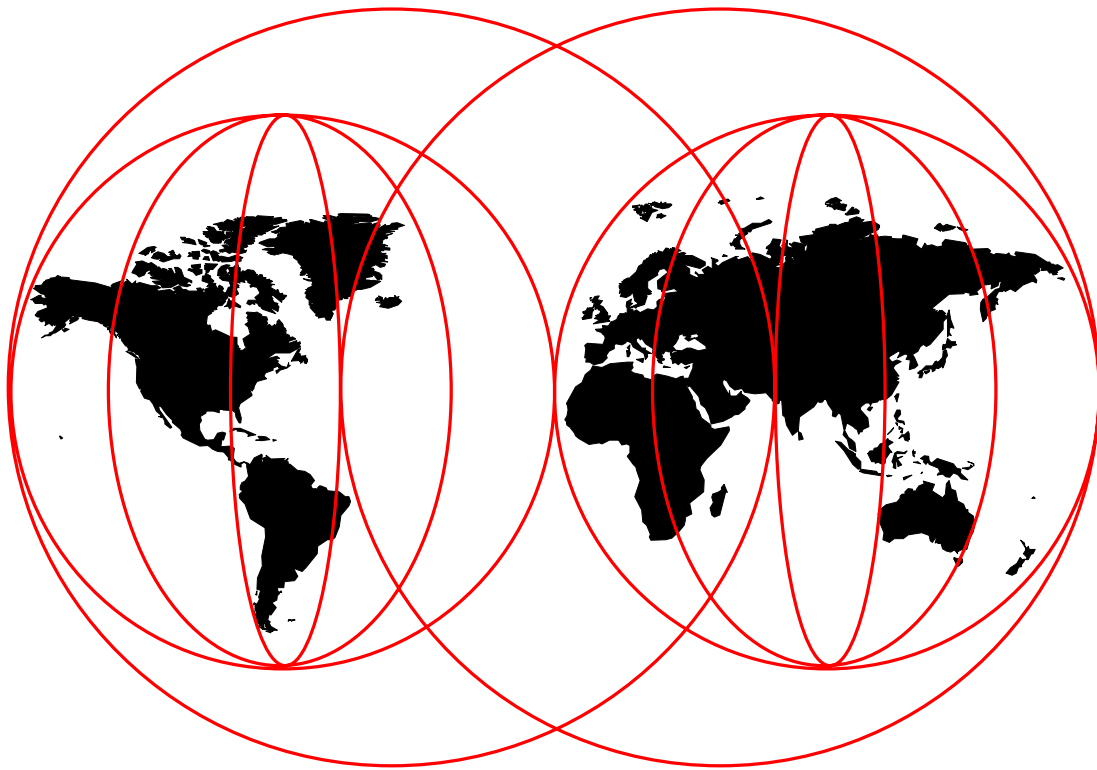


Guarding the Gates Using the IBM eNetwork Firewall V3.3 for Windows NT

Jorge Ferrari, Rik De Waele, Heinz Peter Hippenstiel, Max Stingl



International Technical Support Organization

<http://www.redbooks.ibm.com>



International Technical Support Organization SG24-5209-01

**Guarding the Gates Using the
IBM eNetwork Firewall V3.3 for Windows NT**

July 1999

Take Note!

Before using this information and the product it supports, be sure to read the general information in Appendix I, "Special Notices" on page 627.

Second Edition (July 1999)

This edition applies to Version 3, Release 3 of the IBM eNetwork Firewall for Windows NT, Program Number 5765-C16.

Comments may be addressed to:
IBM Corporation, International Technical Support Organization
Dept. HZ8 Building 678
P.O. Box 12195
Research Triangle Park, NC 27709-2195

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© **Copyright International Business Machines Corporation 1998 1999. All rights reserved.**

Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Contents

Preface	xiii
How to Access Samples from This Book	xiii
The Team That Wrote This Redbook	xiii
Comments Welcome	xiv
Chapter 1. An Introduction to Firewalls	1
1.1 Commercialization of the Internet	1
1.2 What Does "Security" Mean?	3
1.3 Reducing Your Exposure	4
1.4 The Firewall Concept	4
1.4.1 Screening Filter	5
1.4.2 Bastion	5
1.4.3 Dual-Homed Gateway	6
1.5 Firewall Objectives and Firewall Rules	8
1.5.1 Beyond the Firewall: Filtering Content	9
1.6 From Here On...	9
Chapter 2. Introducing the IBM eNetwork Firewall for Windows NT	11
2.1 IBM Firewall for NT Components	11
2.2 IP Filters	12
2.2.1 Objective of the Filters	12
2.2.2 Implementation	12
2.3 Proxy Servers	16
2.3.1 Objectives of the Proxy Servers	16
2.3.2 Implementation	17
2.4 SOCKS Server	17
2.4.1 Objectives of the SOCKS Server	18
2.4.2 Implementation	18
2.5 Domain Name Services	19
2.5.1 Objectives of the DNS Server	19
2.5.2 Implementation	19
2.6 Secure Mail Handling	20
2.6.1 Objectives of the Mail Relay	20
2.6.2 Implementation	20
2.7 Network Address Translation (NAT)	20
2.7.1 Objectives of Network Address Translation (NAT)	21
2.7.2 Implementation	21
2.8 Virtual Private Network	21
2.8.1 Objectives of a Virtual Private Network	22
2.8.2 Implementation	22
2.9 Authentication Methods	22
2.9.1 Objectives of the Authentication Methods	22
2.9.2 Implementation	23
2.10 Logging	23
2.10.1 Objectives of Logging	23
2.10.2 Implementation	23
2.11 Remote Administration	25
2.11.1 Objectives of Remote Administration	25
2.11.2 Implementation	26
2.12 Firewall Setup Wizard	26

2.12.1 Objectives of Setup Wizard	26
2.12.2 Implementation	26
2.13 From Here On...	26
Chapter 3. Planning Your Firewall Installation	29
3.1 Defining a Firewall Security Policy	29
3.1.1 Defining Resources	29
3.1.2 Defining Users	30
3.1.3 Defining Activities	30
3.1.4 Administration	31
3.2 Defining Network Objects	31
3.3 Defining Connections	33
3.4 Proxy Decisions	34
3.5 Network Address Translation Decisions	36
3.6 Naming Services	36
3.7 Mail Services	37
3.8 Virtual Private Network	37
3.9 Alerts, Reports, and Logs	38
3.10 Administration and Firewall Management	39
3.11 Mars Attacks!	40
3.12 IBM eNetwork Firewall for Windows NT Initial Configuration Parameters	41
3.13 From Here On...	41
Chapter 4. Installing the IBM eNetwork Firewall for Windows NT	43
4.1 The Base Operating System	43
4.2 Installing Windows NT Server 4.0	43
4.2.1 Phase 0: Pre-installation	44
4.2.2 Phase 1: Gathering Information about your Computer	45
4.2.3 Phase 2: Installing Windows NT Networking	45
4.2.4 Phase 3: Finishing Setup	46
4.3 Base Operating System Installation Summary	47
4.4 Post-Installation Procedures	47
4.4.1 Checking Your Installation	48
4.4.2 Setting System Resolution	48
4.4.3 Installing the Microsoft DNS Server	49
4.4.4 Installing Service Pack 3 and Post-SP3 Hot Fixes	51
4.5 Enhancing Security on the Base Operating System	52
4.5.1 Physical Security	52
4.5.2 Account Security	52
4.5.3 User Rights Policy	55
4.5.4 Disable Shares	55
4.5.5 Connectivity Test	56
4.6 Installing the Base Firewall	56
4.6.1 Installation Media	57
4.6.2 Logging On	57
4.6.3 Activating Windows NT's IP Forwarding Function	57
4.6.4 Installing the IBM Intermediate Support Device Driver	58
4.6.5 Running the Installation Program	59
4.6.6 Connect the Firewall to the Network	62
4.7 Changes Made by the Hardening Process	62
4.8 Initial Configuration of IBM eNetwork Firewall for Windows NT	63
4.8.1 Prerequisites	63
4.8.2 Checking Your TCP/IP Configuration	64
4.8.3 Starting the Configuration Client	65

4.8.4	Summary of the Configuration Client Main Panel	66
4.8.5	Summary of the Configuration Client Navigation Tree	67
4.8.6	Specifying the Secure and Non-Secure Interfaces	69
4.8.7	Configuring DNS	69
4.8.8	Configuring the Secure Mail Proxy	70
4.8.9	Creating Network Objects	71
4.8.10	Most Common Settings for Security Policy	74
4.9	From Here On...	75
Chapter 5. A Quick and Basic Configuration: A Case Study		77
5.1	Case Study: Faire.Com	77
5.2	Planning	78
5.2.1	Faire.Com's Intranet/Internet Layout	78
5.2.2	Internet Security Policy	79
5.2.3	Network Details	80
5.3	Configuring Peripheral Services	81
5.3.1	Configuring the External Name Server	81
5.3.2	Configuring the Internal Name Server	82
5.3.3	Configuring the Internal Mail Server	89
5.3.4	DNS Client Configuration	90
5.3.5	Default Gateway Configuration	91
5.4	Configuring IBM eNetwork Firewall for Windows NT	92
5.4.1	Configuring the Interfaces	92
5.4.2	Configuring Name Services	93
5.4.3	Configuring for Secure Mail Services	94
5.4.4	Creating Network Objects	95
5.4.5	Configuring Security Policies	96
5.4.6	Configuring SOCKS Services	97
5.4.7	Reviewing and Activating the Connections	100
5.4.8	Configuring Alerts	101
5.4.9	Log Archiving	104
5.4.10	Configuring the Default Gateway	105
5.5	Configuring Clients to Use the Firewall	105
5.5.1	Configuring DNS and the Default Gateway	105
5.5.2	Configuring Web Clients	105
5.6	From Here On...	106
Chapter 6. Firewall Setup Using the Wizard		109
6.1	Wizard Objectives	109
6.2	Wizard Sample Configuration	109
Chapter 7. Coming to Grips with IP Packets		127
7.1	An Introduction to IP Packets	127
7.2	An Introduction to ICMP Packets	129
7.2.1	Echo and Echo Reply Messages	130
7.2.2	Destination Unreachable Message	130
7.2.3	Source Quench Message	131
7.2.4	Redirect Message	131
7.2.5	Time Exceeded Message	135
7.2.6	Parameter Problem Message	135
7.2.7	Time Stamp and Time Stamp Reply Message	135
7.2.8	Information Request Message	136
7.2.9	Address Mask Request and Address Mask Reply Messages	136
7.2.10	Router Advertisement and Router Solicitation Message	136

7.2.11 Domain Name Request and Domain Name Reply Messages	137
7.2.12 Traceroute Message	137
7.2.13 ICMP Message Summary	138
7.3 An Introduction to TCP Packets	138
7.4 Use and Abuse of TCP Ports	140
7.4.1 Source Porting	140
7.4.2 Stealth Scanning	141
7.5 An Introduction to UDP Packets	141
Chapter 8. IBM Firewall Rule Base	143
8.1 Rule Base	143
8.1.1 Connections	143
8.1.2 Objects and Groups	144
8.1.3 Services	145
8.1.4 Rules	148
8.1.5 Names and Descriptions Convention	150
8.2 Rule Base Design	151
8.2.1 Standard Connections	151
8.2.2 Non-Standard Connections	155
8.2.3 Rule Base Activation	159
8.2.4 Performance	160
Chapter 9. Examples of Rules for Specific Services	161
9.1 What Services Should You Provide?	161
9.2 Connection Rules that Should Always Be Present	162
9.2.1 Rules to Block Attempts at IP Address Spoofing	163
9.2.2 Rules to Control ICMP Message Flow	165
9.2.3 Rule to Isolate Private Networks from the Internet	165
9.2.4 Rule to Protect the SOCKS Service on the Non-secure Interface	166
9.2.5 Rules to Protect from Loopback Network	167
9.2.6 Broadcast	168
9.2.7 Routed Traffic	169
9.3 Telnet	170
9.3.1 Telnet From Firewall to Secure Network	171
9.3.2 Telnet Using Proxy	172
9.3.3 Telnet Using SOCKS	173
9.4 FTP: File Transfer Protocol	175
9.4.1 Normal Mode	175
9.4.2 Passive Mode	175
9.4.3 FTP from Secure Network to Non-secure Network Using Proxy	176
9.4.4 FTP from Secure Network to Non-secure Network Using SOCKS	179
9.4.5 Normal Mode FTP from Firewall to Secure Network	180
9.5 VPN: Virtual Private Network	181
9.5.1 Data Traffic Connection	182
9.5.2 Encapsulation Connection	183
9.6 SMTP: Simple Mail Transfer Protocol	184
9.7 DNS: Domain Name Server	188
9.8 NNTP: Network News Transfer Protocol	189
9.9 HTTP - World Wide Web Sessions	192
9.9.1 Possible Scenarios	192
9.9.2 SOCKSified Client	193
9.9.3 Internal HTTP Proxy Server Using SOCKS	194
9.9.4 HTTP Proxy Server in the Firewall	195
9.9.5 Internal HTTP Proxy Server Using IP Forwarding	195

9.9.6 External HTTP Proxy Server	195
9.10 SSL: Secure Sockets Layer	196
9.11 S-HTTP	197
9.12 Gopher	197
9.13 Lotus Notes and Domino	199
9.14 ident	201
9.15 Traceroute	202
9.15.1 Traceroute from the Firewall	203
9.15.2 Traceroute from Internet to the Firewall	204
9.16 Network Management Sessions	205
9.17 Archie	207
9.18 WAIS	208
9.19 HOD: Host On-Demand	210
9.19.1 HOD Server Using IP Forwarding	210
9.19.2 HOD Server Using SOCKS	213
9.20 Filtering Specific ICMP Messages	216
9.20.1 Consolidated ICMP Connection	216
9.21 Other Protocols	218
9.22 SecurID by Security Dynamics	219
9.23 Using the Security Policy Panel	219
Chapter 10. Remote Administration Configuration	221
10.1 Setting Up the Configuration Server	221
10.1.1 The Configuration Service Port Number	221
10.1.2 The Client Configuration File fwcfgsrv	222
10.1.3 Enabling Remote Configuration Client Connections	223
10.2 Enabling Encrypted Remote Configuration	226
10.2.1 Creating a Key File with mkkf	226
10.2.2 A Sample mkkf Session	226
10.2.3 Changes in the Configuration File	230
10.3 Creating Administrators	231
10.3.1 The Primary Administrator Account	231
10.3.2 Creating a Firewall Administrator Account	235
10.3.3 Administration Functions	236
10.4 Setting Up the Configuration Client	237
10.5 Changing Port Numbers	240
10.5.1 Creating New Rules and Connections	241
10.5.2 Logging in Using the New Port Numbers	244
10.6 From Here On...	245
Chapter 11. Domain Name Service (DNS)	247
11.1 How Does DNS Work?	247
11.2 Some Attacks on DNS	251
11.2.1 Cache Poisoning	251
11.2.2 UDP Denial-of-Service Attack	252
11.3 Implementation of DNS in IBM Firewall	252
11.3.1 The Firewall Hides the Information about Internal Hosts	252
11.3.2 The Firewall Permits DNS Resolution of External Hosts	252
11.4 Examples of DNS Configuration	254
11.4.1 Configuration of the Internal DNS	255
11.4.2 Configuration of the External DNS	259
11.4.3 Configuration of DNS in Firewall	263
11.4.4 Testing The DNS Configuration	268

Chapter 12. Mail Handling	269
12.1 Functionality of Secure Mail Proxy	269
12.1.1 SMTP Commands	269
12.1.2 Multiple Secure Mail Servers	270
12.1.3 Host Name and Domain Name Rewriting	270
12.2 How Does Secure Mail Proxy Work?	271
12.2.1 The Overflow Server	271
12.2.2 Incoming Mail Flow	272
12.2.3 Outgoing Mail Flow	275
12.3 Configuring Mail Handling	278
12.4 Configuring an Overflow Server	283
12.5 Tailoring the Secure Mail Proxy	284
12.6 Mail Handling Examples	284
12.6.1 Incoming Mail to the Secure Network	286
12.6.2 Outgoing Mail to Nonsecure Network	287
12.7 Mail Log	289
Chapter 13. Configuring Proxy Services and SOCKS	293
13.1 User Administration	294
13.1.1 Types of Users	294
13.1.2 Types of User Databases	294
13.1.3 User Authentication Schemes	295
13.1.4 Configuring Proxy Users	296
13.1.5 Configuring the Default User	301
13.1.6 Configuring for NT Logon Password Authentication	302
13.2 Managing Proxy and SOCKS Services	306
13.2.1 Stopping and Starting Services	307
13.2.2 Disabling and Enabling Services	307
13.3 Using the Telnet and FTP Proxy	308
13.3.1 Configuring Proxy Services	308
13.3.2 Example Proxy Connections	311
13.4 Using the Transparent Proxy	313
13.4.1 Configuring Transparent Proxy Services	313
13.4.2 Example Transparent Proxy Connections	314
13.5 Idle Proxy Connections	315
13.5.1 Configuring Idle Proxy Connections	315
13.6 Using the HTTP Proxy	316
13.6.1 Authentication Levels	316
13.6.2 Configuring HTTP Proxy Services	316
13.6.3 Example HTTP Proxy Connection	321
13.7 Using the SOCKS Server	324
13.7.1 User Authentication Modes	325
13.7.2 Configuring SOCKS Services	326
13.7.3 Advanced Configuration	331
13.7.4 Example SOCKS Connections	335
13.7.5 SOCKS Client Services	336
Chapter 14. Network Address Translation	339
14.1 Translation Mechanism	339
14.2 Configuration of NAT	341
14.3 How to Configure Routing when Using NAT	348
14.3.1 The Registered NAT IP Address is in a Separate Subnet	348
14.3.2 The Registered NAT IP Address is in the Same Subnet	348
14.3.3 Maximum Transmission Unit	349

14.4	Timeout Value	350
14.5	Example Configurations for Using NAT	351
14.5.1	Using NAT to the Internet	351
14.5.2	Mapping a Server	352
14.5.3	Connecting two Networks with Same IP Addresses	353
14.6	NAT and Virtual Private Networks (VPNs)	355
Chapter 15. Virtual Private Network		357
15.1	Secure IP Tunnel Standards - Interoperability	357
15.2	Operation of the Secure Tunnel	358
15.3	Implementing the Secure IPSEC Tunnel	360
15.3.1	Adding the Tunnel Definition in One Node	360
15.3.2	Export the Tunnel Definition to a File	364
15.3.3	Import the Tunnel Definition in the Partner Node	367
15.3.4	Activate/Deactivate the Tunnel at Both Ends	369
15.3.5	Using Static Filter Rules	371
15.3.6	Reactivate Tunnel When Life Time has Expired	375
15.3.7	Summary	375
15.4	Authentication and Encryption Examples	376
15.4.1	Authentication Example	377
15.4.2	Encryption Example	378
15.5	Virtual Private Network Scenarios	380
15.5.1	Tunnel between Two IBM eNetwork Firewalls for Windows NT	380
15.5.2	Virtual Private Network between IBM eNetwork Firewall for Windows NT and AIX V4.3	391
15.5.3	Virtual Private Network between IBM eNetwork Firewall for Windows NT and OS/390	395
15.5.4	VPN between IBM eNetwork Firewall for Windows NT and OS/400 V4R4	404
Chapter 16. Logging, Monitoring, and Reporting		409
16.1	Configure Logging	409
16.1.1	Logging Priority Levels	409
16.1.2	Log Facilities	410
16.1.3	Manage Log Facilities	410
16.1.4	Archive Log Files	413
16.1.5	Manage the Admin Audit Log Facility	417
16.1.6	Examine the Firewall Log Files	417
16.1.7	Configure Logging Sources	418
16.2	Monitoring and Alerts	420
16.2.1	Log Monitor Thresholds	420
16.2.2	Alert Messages	421
16.2.3	Alert Message Delivery Methods	422
16.2.4	Log Monitor Administration	422
16.2.5	Configure Threshold Monitors	424
16.2.6	Configure Delivery Monitors	426
16.2.7	Pager Setup	431
16.3	Log File Formats	433
16.3.1	Firewall Log Format	434
16.3.2	Alert Log Format	435
16.3.3	Admin Audit Log Format	435
16.4	Building Reports	436
16.4.1	Report Utilities	437
16.5	Using SOCKS Traffic Monitor	440

16.5.1	Connection for Remote Usage	441
16.5.2	Starting the SOCKS Traffic Monitor	444
16.5.3	Monitor Traffic	444
16.5.4	Edit Traffic Monitor	447
Chapter 17. Configuring the IBM Firewall for Three Adapters		449
17.1	Considerations with Two Non-secure Interfaces	450
17.2	Configuration Example	451
17.2.1	Installing the Third Adapter	452
17.2.2	Adding Network Objects	454
17.2.3	Configuring Connection from Internet to Domino2 Server	456
17.2.4	Connection from Domino1 Server to Domino2 Server	465
Chapter 18. Testing The Configuration		475
18.1	Introduction	475
18.2	Testing	475
18.3	Network Security Auditor	494
18.4	SATAN	495
Chapter 19. Backup, Restoration and Migration of the IBM Firewall		497
19.1	The IBM Firewall Configuration Files	497
19.1.1	The Configuration Map File	497
19.1.2	Connection Configuration Files	497
19.1.3	Application Proxies and SOCKS Configuration Files	498
19.1.4	Mail Settings File	498
19.1.5	DNS Settings Files	499
19.1.6	Virtual Private Network	499
19.1.7	Network Address Translation Configuration File	499
19.1.8	Logging and Alert Configuration Files	499
19.1.9	Modem and Pager Configuration Files	500
19.1.10	Configuration Server Files	500
19.2	Backing Up	500
19.3	Restoring	501
19.4	Migration	501
Chapter 20. SecureWay Network Dispatcher		503
20.1	The Basics of ND	503
20.2	Technical Background	504
20.2.1	Installation and Configuration of the ND	504
20.2.2	Configuration of the Firewall	507
20.2.3	How it works	508
20.2.4	Load Balancing with Different Services	508
20.2.5	The Wildcard Port	512
20.2.6	The Wildcard Cluster	512
20.3	ND Advanced Usage: NAT One-to-Many Simulation	513
20.4	High Availability	515
Chapter 21. Integration with Other Products		517
21.1	IBM DB2	517
21.1.1	Create the Initial Database	517
21.1.2	Import the Log Data	519
21.1.3	Sample Queries	520
21.2	Aventail AutoSOCKS	521
21.2.1	Major Features	522

21.2.2	Installing Aventail AutoSOCKS	523
21.2.3	Configuring AutoSOCKS	525
21.2.4	AutoSOCKS Utilities	528
21.3	IBM Web Traffic Express	529
21.3.1	Using IBM Web Traffic Express on Firewall	530
21.3.2	Using IBM Web Traffic Express on a Separate Host	544
21.4	MIMESweeper	550
21.4.1	Installation	552
21.4.2	Configuration of MIMESweeper on the SMTP Gateway	555
21.4.3	Test the Configuration	562
21.5	SecurID Authentication	563
21.5.1	How the ACE/Server Implements Strong Authentication	564
21.5.2	Configuring the ACE/Server	564
21.6	TME 10	576
21.6.1	Installation	578
21.6.2	Configuration of TME 10 Enterprise Console	581
21.6.3	Test the Configuration	594
Appendix A. Security Issues Old and New		597
A.1	Network Vulnerability	597
A.1.1	Denial-of-Service Attacks	597
A.1.2	Snooping	598
A.1.3	Man-in-the-Middle	598
A.2	Application Vulnerability	599
A.2.1	Java, Javascript, and Active X	599
A.2.2	Viruses and Trojan Horses	599
A.2.3	Dumb, Dumb Users!	600
A.3	Windows NT Vulnerability	600
A.4	Traditional Vulnerabilities	600
A.5	From Here On...	601
Appendix B. Planning Forms		603
Appendix C. Recommended ICA Message Tags		607
Appendix D. Protocol Numbers		609
Appendix E. Well-Known Port Numbers		613
Appendix F. ICMP Message Types and Codes		615
Appendix G. List of IBM Firewall Rules		619
Appendix H. Related Publications, Web Sites, and Mailing Lists		623
H.1	International Technical Support Organization Publications	623
H.2	Redbooks on CD-ROMs	623
H.3	Other Publications	624
H.4	Web Sites	624
H.5	Mailing Lists	625
Appendix I. Special Notices		627
How to Get ITSO Redbooks		629
IBM Redbook Fax Order Form		630

Index	631
ITSO Redbook Evaluation	633

Preface

This redbook describes the implementation of the IBM eNetwork Firewall V3.3 for Windows NT for enforcing security at the boundaries of TCP/IP networks. It contains detailed information on this new IBM firewall, as well as basic and advanced configuration techniques. It also gives several helpful tips on the Windows NT security and general Internet security.

The chief feature of this redbook is its numerous examples. These examples cover all the major capabilities of the IBM firewall: IP filters; proxy and SOCKS services; logging, monitoring, and alerts; DNS; mail; and remote configuration. It goes two steps further: first, by covering configuration examples of the systems around the firewall; and second, by discussion of various third-party products that enhance firewall functionality.

This redbook will help you to plan, install, implement, and manage the IBM eNetwork Firewall V3.3 for Windows NT. The coverage is sufficiently wide as to satisfy the needs of both novice and advanced users. However, some hands-on experience with Windows NT and a working knowledge of TCP/IP would be most helpful.

This is an update of the redbook on IBM eNetwork Firewall V3.2 for Windows NT; the changes and additions in this edition are marked with revision bars.

How to Access Samples from This Book

Code referenced in this document is available through the Internet.

For the files referenced in this book, point your Web browser to <http://www.redbooks.ibm.com>, click on **Additional Materials**, and click on **SG245209**.

The Team That Wrote This Redbook

This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization, Raleigh Center.

Jorge Ferrari is a Network Security Specialist in the International Technical Support Organization, Raleigh Center. Before working in the security area, he was a specialist in network design and capacity planning. He holds a degree in Electronic Engineering from the Universidad de Buenos Aires, Buenos Aires, Argentina.

Rik De Waele is a member of the Networking support and services group of IBM Belgium. He has been working with IBM for the last four years. The first two years he did support for TCP/IP on the PC platform; later on he moved to the network security area. Before entering IBM, he worked for seven years as a software developer.

Heinz Peter Hippenstiel is a security and network consultant with HAITEC AG, Germany. His areas of expertise cover Windows NT and UNIX; he is a Microsoft

Certified Professional. He holds a distinction as Staatlich Geprüfter Informatiker from the Akademie für Datenverarbeitung in Böblingen, Germany.

Max Stingl is a Certified Firewall and AIX Teacher in IBM Germany. He has worked for IBM for 25 years, primarily with IBM midrange systems. Max has been a system engineer for most of his career. His area of expertise includes several years in robotics and computer-aided manufacturing. He joined IBM South Africa for three years working on COPICS before he went back to Germany. Currently, his focus is on firewall classes for Windows NT and AIX, network security and Internet technologies.

The authors of the first edition of this redbook, also under advisor Jorge Ferrari, were:

Dominique G. M.Cimafranca, IBM Philippines
Marcello B. Gomes, IBM Brazil
Andre Jenie, IBM Indonesia
Sven Müssig, IBM Germany

Thanks to the following people for their invaluable contributions to this project:

Andrew Yeomans, IBM UK
Marco Pistoia, ITSO Raleigh
Marcela Adan, ITSO Rochester
Karl Wozabal, ITSO Raleigh
Martin Murhammer, ITSO Raleigh
Corinne Letilley, IBM Global Services Canada

Gordon Arnold
Kurt Deitrick
Kevin Dunphy
Martin Gramlich
Chris Higgins
Tai W. Nam
Mark North
Bill Serencsics
Susanne Vergara
from the IBM Firewall development team

Randy Craig
from IBM Human Interface Group

Chris Henrion
Steven Song
from IBM Integrated Network Marketing and Test Services

Comments Welcome

Your comments are important to us!

We want our redbooks to be as helpful as possible. Please send us your comments about this or other redbooks in one of the following ways:

- Fax the evaluation form found in “ITSO Redbook Evaluation” on page 633 to the fax number shown on the form.
- Use the online evaluation form found at <http://www.redbooks.ibm.com/>
- Send your comments in an Internet note to redbook@us.ibm.com

Chapter 1. An Introduction to Firewalls

This redbook is all about Internet and network security, in particular, about security as it can be implemented and enforced with the IBM eNetwork Firewall for Windows NT.

Firewalls have been in existence for quite some time now, evolving in form and function with the needs and requirements of organizations participating in the Internet. With the explosive growth of the Internet, firewalls have become a growth industry in themselves. Unfortunately, this rapid growth has far outpaced the efforts of IT managers to keep abreast of firewalls and security issues.

Our objective in this chapter is to give you a capsule summary of the importance of Internet security and how you can use firewalls to your advantage in minimizing the dangers that lurk out there. We begin with a discussion of the growing importance of the Internet to your business followed by a brief section on general security issues. In the last sections, we cover current firewall architectures and the general principles of firewall design, as a prelude to an in-depth look at the characteristics of the IBM Firewall in Chapter 2, "Introducing the IBM eNetwork Firewall for Windows NT" on page 11.

The material in this chapter borrows heavily from the equivalent chapter of *Protect and Survive: Using IBM Firewall 3.1 for AIX, SG24-2577-02*.

1.1 Commercialization of the Internet

The Internet is a worldwide IP network that links many different organizations. The Internet is not a centralized organization but a collection of different networks from various sources, governmental, educational and commercial. There is a single administration for managing IP addresses and naming domains, the *Internet Assigned Numbers Authority*, but no other central administrative function. Internet routing is done by many Internet providers, government departments and private service companies who establish connections among themselves and build the base of the network. Organizations connected to the Internet are usually bound to one provider and so may communicate with any other connected organization across the inter-provider routes.

In the past, the Internet was suited to data processing and scientific people, using unfriendly interfaces such as Telnet or FTP. Today, everybody who knows how to use a keyboard and click a mouse can use a very friendly Web browser client to navigate the World Wide Web. The Web has had a profound effect on both the volume of traffic carried by the Internet and also on traffic patterns, since by following hypertext links, a user may rapidly establish connections with many different server machines. This means that:

1. There are many more sessions being set up, as a single client is likely to communicate with many more servers than in the past.
2. There are many more client machines, because the interface is familiar and easy to use and many organizations have rushed to provide Internet access to almost anyone.
3. There are many more servers, partly because many companies and organizations see the Internet as a way to disseminate information and partly

because low-cost service providers have allowed individuals and small organizations to provide their own Web sites.

So successful has the World Wide Web been, that to many people today it *is* the Internet.

The Internet today is metamorphosing from a vehicle for academic discussion and the sharing of research data, into a global information resource for private and commercial use.

Many companies feel that the world of the Internet promises opportunities, in several forms:

- Inter-Company Communications

The use of e-mail has radically changed the way that companies operate internally, by speeding communications, increasing the reach of corporate announcements and introducing a new egalitarianism (that is, people will send e-mail to managers and directors to whom they would not dream of telephoning or writing). Mail access via the Internet brings the same sort of benefits to inter-enterprise communications as well as a way for individuals to communicate.

- Marketing, Sales and Support

The Internet offers a huge marketplace for promoting products and services to a targeted audience. Some companies have flooded the net in a crass and insensitive way, generating a lot of anger among the Internet community. However, as the Internet becomes increasingly commercialized and with the advent of new access methods such as the World Wide Web, product promotion and support has become an increasingly important portion of network traffic.

- Productivity and Competitive Edge

No matter what subject you are interested in, you will find information about it *somewhere* in the Internet. Providing rapid access to information sources can give a major productivity boost. Also access to data allows employees to make more informed decisions and to better explore business opportunities.

- Conducting Business

Until recently, business on the Internet was inhibited by the absence of good security standards. Several developments have addressed this problem. Probably the most far-reaching is the global availability of the Secure Sockets Layer (SSL) by Netscape Corporation. This allows Web transactions to take place with good privacy and authentication, using public-key encryption techniques. Many other security technologies have been widely used for some time, such as Pretty Good Privacy (PGP) for secure mail transfer, but unlike SSL they required some technical knowledge from the user. SSL is a generic solution to the problem of ensuring privacy, but it is not an answer to everything. For example, the Secure Electronic Transactions (SET) standard will enable secure credit card payments to become a reality. In parallel with these new technologies, there has been an easing of the U.S. cryptography export restrictions.

It is clear that there will be a lot more commercial activity with security developments such as these that allow safe transfer of private data, credit card numbers, etc.

This commercial interest in the Internet is fuelled by and also a cause of the explosive growth in Internet hosts and traffic.

So much for the opportunities; what about the dangers? There has been much press comment about crackers on the Internet, some of which has over-sensationalized their exploits. The threat that crackers have posed to date is less severe than the popular image. Most of the publicized incidents have been perpetrated as a means to grab attention and not with any real malicious intent. However, the ingenuity of the crackers in defeating system defenses has *not* been overstated. The really frightening prospect, therefore, is if one of these wily crackers has some personal or commercial reason to do you harm.

A Word About Crackers

Here is the definition of cracker taken from the Free Online Dictionary of Computing (FOLDOC) at <http://wombat.doc.ic.ac.uk/foldoc/>:

Cracker:

<jargon> An individual who attempts to gain unauthorised access to a computer system. These individuals are often malicious and have many means at their disposal for breaking into a system. The term was coined ca. 1985 by hackers in defence against journalistic misuse of "hacker". An earlier attempt to establish "worm" in this sense around 1981--82 on Usenet was largely a failure.

1.2 What Does "Security" Mean?

There is always a trade-off to be made between making a computer secure and the function it can provide. In the extreme case, the most secure computer is one that is turned off. With networked computers the problem is compounded since the communication channel itself is open to attack. We can characterize attacks in two ways:

1. Passive attacks: Tapping or tracing the communications. These are very difficult to detect. You should assume that someone is eavesdropping on every communication you send across the Internet.
2. Active attacks: That is, a cracker is trying to take over your machines. Even if you are certain that your own machines have not been compromised, you cannot be certain about the machines at the other end of the connection. Realistically you have to extend your circle of trust to some of those machines, or else you will not use the Internet at all.

It seems that once you start getting paranoid about computer security you can reach a point where nothing seems safe anymore. Is this justifiable? After all, we don't (normally) worry about people tapping our telephone conversations or reading our mail, and we happily send credit card numbers, private messages, gossip and scandal using those media. The difference with the Internet is that the carrier is not a regulated, well-defined entity. In fact you have no idea whose computers your message passes through on the way to its destination.

1.3 Reducing Your Exposure

This book concentrates on protecting the points of entry, where the Internet meets your private network. However, we will also consider authentication and encryption of data passing across the Internet, using secure tunnels and alternatives for secure terminal emulation.

The any-to-any connectivity of the Internet can, alone, give you many security problems. You will need to protect your own private data and also protect access to the machines inside your private network against abusive external use.

The first step to achieving this is to limit the number of points at which the private network is connected to the Internet. The best configuration is where the private network is connected by just one gateway. If you only have this unique path, you have gained a *choke point*, where you can exercise control over which traffic to allow into and out of the Internet. We call this gateway a firewall. When considering the details of your firewall it is important to keep in mind that the whole concept relies on the fact that it is the only gateway. If an attacker can find an alternative, uncontrolled access point, your firewall effort has been wasted. Every point of entry must be subjected to the same rigorous standards of security.

Depending on what your security requirements are and how much money you have, you can develop different firewall strategies. The most important advice is to keep the strategy simple.

1.4 The Firewall Concept

To understand how a firewall works, consider this example. Imagine a building where you want to restrict access and to control people who enter in. You define in the architecture of the building a single lobby as the only entrance point. In this lobby, you have some receptionists to welcome, some security guards to watch over, some video cameras to record, and some badge readers to authenticate people who enter the building.

This works very well to control a private building. But imagine that a non-authorized person succeeds in entering, no matter how. To protect the building against any actions from this person is more difficult. However, if you supervise his or her movements, you at least have a chance to detect any suspicious behavior and repair any damage.

When you are defining your firewall strategy, you may think it is sufficient to prohibit everything that presents a risk for the organization and allow the rest. However, because of new attack methods, you may not be able to prevent every attack and, as in the case of the building, you need to monitor for signs that somehow your defenses have been breached. Generally, it is much more damaging and costly to recover from a break-in than to prevent it in the first place.

In the case of the firewall, the classic solution uses a screening router. Although that is not sufficient today to ensure security, it is still the starting point for firewall defenses. A better strategy is to permit only the applications you have tested and have confidence in. If you follow this strategy, you have to exhaustively define the list of services you must run on your firewall. Each service is characterized by the direction of the connection (from in to out, or out to in), the list of users authorized,

the list of machines where a connection can be issued, and perhaps the range of time of day you authorize this service.

— If You Want to Refresh Your Knowledge about TCP/IP —

In the following sections we assume a fair understanding of IP protocols, addressing, and router configurations. If you want to increase your understanding of TCP/IP, we recommend you read *TCP/IP Tutorial and Technical Overview*, GG24-3376. Also, if you encounter words or phrases that you do not understand, you may find an explanation in the glossary.

1.4.1 Screening Filter

The first and most commonly used strategy is to separate the private IP network from the Internet by inserting a router between them, as shown in Figure 1.

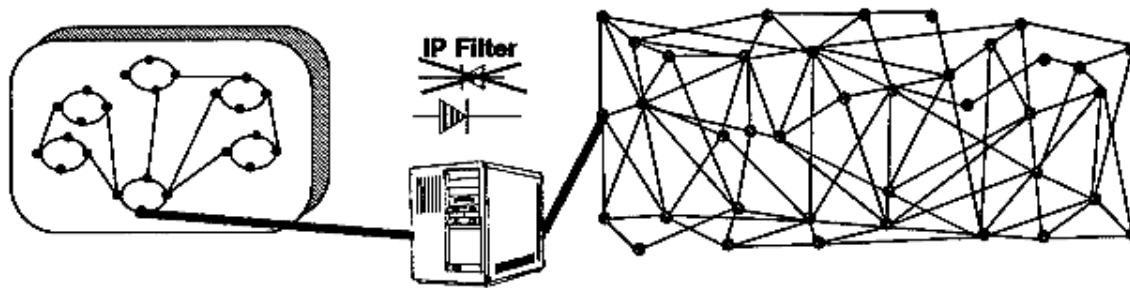


Figure 1. Screening Filter

This router filters all IP packets passing through and is called a screening filter. This way you can prevent access to machines or to ports in the private network and also do the reverse: prevent an inside machine from accessing the Internet. But if you do this, there is no way to control what's happening at the application layer. That is, you may want to allow one type of traffic across the gateway but not another. You could manage this at the application host itself, but the more machines on which you have to impose controls, the less control you have. Nonetheless a screening filter is a very useful tool to use in conjunction with other tools as a security building block.

1.4.2 Bastion

A bastion is a machine placed between the secure and non-secure network where the IP forwarding is broken, which means no IP packet can go through this machine. As the routing is broken, the only place from which you can access both networks is the bastion itself. Therefore, only users who have an account on the bastion, with a double identification (one for the bastion and one for the remote host), can use services on both the networks, as shown in Figure 2 on page 6.

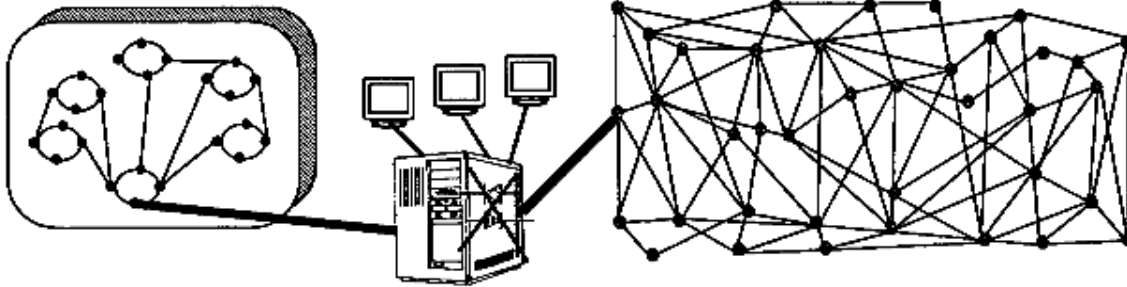


Figure 2. Bastion

This has some disadvantages, because the bastion may have to support many users. It is important to enforce good password control here since if a cracker manages to break into a user ID he or she can then impersonate the user and get into the private network. Besides this security point, supporting a great number of users will require a big machine. To avoid having users logged in to this machine and to reduce load on the machine, application proxies and SOCKS are now being used (see Chapter 13, "Configuring Proxy Services and SOCKS" on page 293).

1.4.3 Dual-Homed Gateway

One good solution is to combine a screening filter and a bastion, as shown in Figure 3.

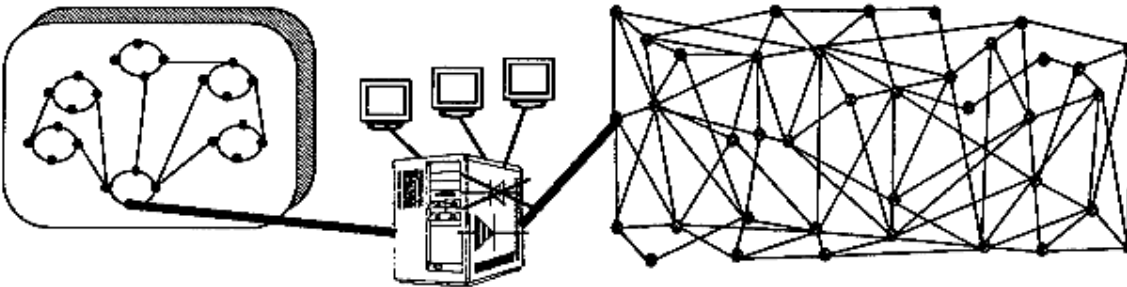


Figure 3. Dual-Homed Gateway

In this case, you can protect the dual-homed gateway from external attacks with filtering. For example, if you forbid external access to the telnet daemon, you reduce the threat of an external attack. If you have some nomadic machines that are hosted outside but need to connect to hosts inside the private network, you can limit the exposure by using a proxy server and perhaps using smart card authentication techniques.

The IBM eNetwork Firewall for Windows NT comes as a configuration similar to a dual-homed gateway,

1.4.3.1 Bastion Behind a Screening Filter

A better solution is to use the same solution as above but use two machines as shown in Figure 4 on page 7.

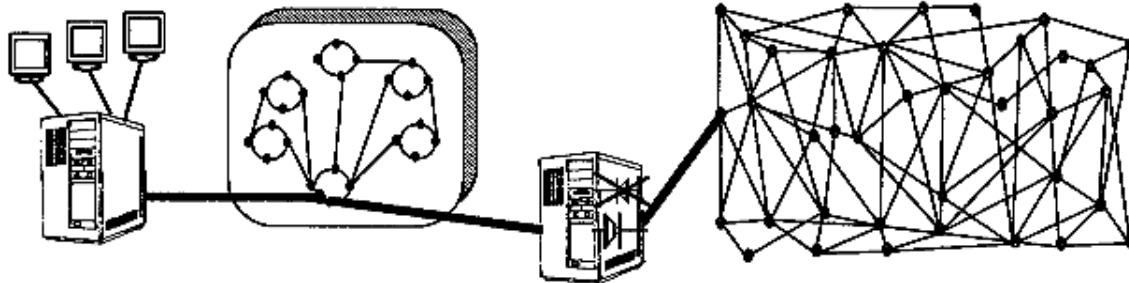


Figure 4. Bastion Behind a Screening Filter

In this configuration, the bastion is protected from external attack by the screening filter. The filter is a very simple router not running any applications or services.

1.4.3.2 Screened Subnet

A further development of this is to use the subnetwork between the screening filter and the bastion as a site for application services. This is increasingly common, as organizations want to provide machines that are widely available (such as Web servers) but still have strong protection for their private network. The screening filter provides some protection for the service machines, without unduly limiting access. A possible example of this is shown in Figure 5. This network is composed of two screening filters and one or several bastions. When you start considering this sort of solution, the cost becomes a major factor since, for reasons of integrity, each component in the design should ideally be a dedicated machine.

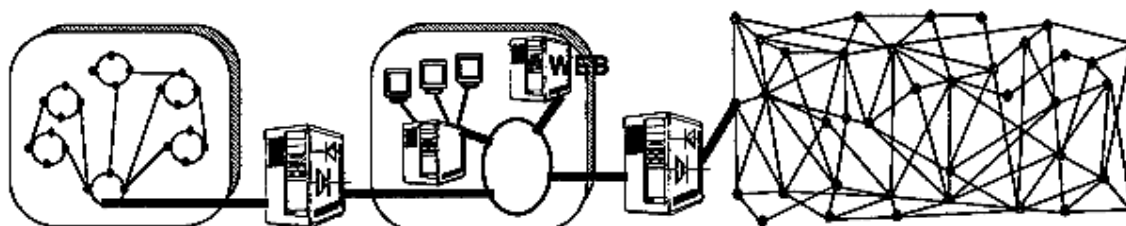


Figure 5. Screened Subnet

The screened subnet is often referred to as the *demilitarized zone* (DMZ).

A cheaper alternative to using a full-blown subnet as DMZ is to use three adapters on the firewall. A DMZ can be created on one of the adapters, an area where secure traffic never flows in and where we can limit activities with the firewall. Figure 6 on page 8 gives you an idea of what this looks like. Chapter 17, "Configuring the IBM Firewall for Three Adapters" on page 449 discusses how to implement this with the IBM Firewall.

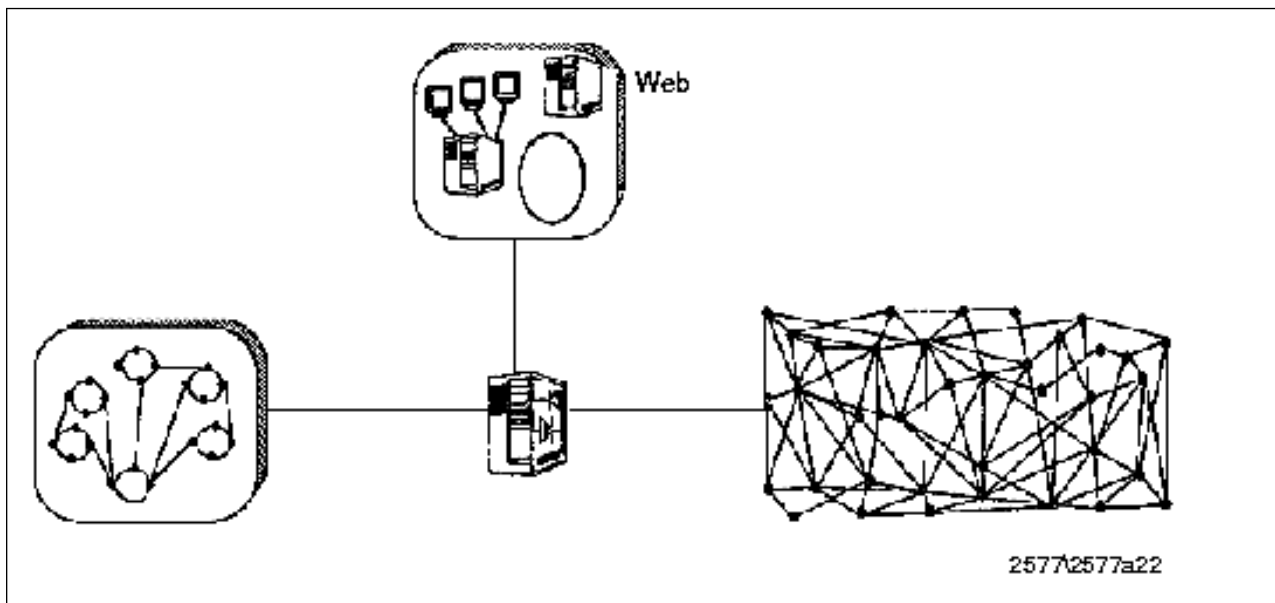


Figure 6. Screened Subnet Implemented with Three-Adapter Firewall

1.5 Firewall Objectives and Firewall Rules

We have seen that there are a number of ways to configure a firewall, depending on the size of your organization and what you are trying to achieve. Before we look in detail at how to use the IBM eNetwork Firewall for Windows NT to achieve some of these configurations, let us state some high-level objectives and rules.

There are a number of objectives that are common to all firewall cases:

- You want to only allow traffic to flow that you have determined is safe and in your interest.
- You want to give away a minimum of information about your private network.
- You want to be able to keep track of firewall activity and be notified of suspicious behavior.

These common objectives translate into a number of rules that you should always keep in mind:

- *Anything that is not explicitly permitted should, by default, be denied.*

What this means is that when you set up your firewall you should be able to state exactly what traffic you want to pass through it. It should not be possible for any other traffic to pass.

- *You should keep outside users out of your internal network wherever possible.*

Even if you are providing a legitimate service for outsiders to use, you should not trust them. If possible such services should be placed outside the firewall (possibly within a DMZ), isolated from your internal systems.

- *You should do thorough auditing and logging.*

You should assume the worst, that at some time your systems will be compromised by a cracker. At this point you need good logging functions to

allow you to detect the cracker, retrace his or her movements, and prevent further damage.

These objectives will find their way into your security policy. Chapter 3, "Planning Your Firewall Installation" on page 29 is devoted to developing a security policy involving the firewall.

1.5.1 Beyond the Firewall: Filtering Content

Firewall technology must wage a continuous battle against the ingenuity of the cracker. This means continuing to be vigilant in preventing misuse of legitimate IP network services and new security holes.

One area that has become part of this battlefield in recent times is the potential for exploitation of "smart" client function. As the Web browser becomes the complete do-everything client, Web-based applications start to rely on client side execution of programs, using techniques such as Java, ActiveX controls and other plug-in function.

Several tools have come out on the market for implementing these functions. These tools are not firewalls in themselves, but can work together with the firewall to provide the content-screening functionality you want. 21.3, "IBM Web Traffic Express" on page 529 and 21.4, "MIMESweeper" on page 550 discuss two related products that work with the IBM Firewall.

1.6 From Here On...

This chapter has discussed the general concepts of firewalls and Internet security.

- To learn more about the IBM eNetwork Firewall for Windows NT, read Chapter 2, "Introducing the IBM eNetwork Firewall for Windows NT" on page 11.
- To learn more about developing a security policy prior to configuring your firewall, read Chapter 3, "Planning Your Firewall Installation" on page 29.

Chapter 2. Introducing the IBM eNetwork Firewall for Windows NT

The IBM eNetwork Firewall for Windows NT is the latest in a long line of firewall products from IBM. It traces its ancestry to the original Internet firewalls that have been protecting IBM's intranet since 1987 and to the IBM NetSP Secured Network Gateway released commercially in December 1994. A short summary of its lineage:

Table 1. A Brief History of IBM Firewalls

Release Date	Name
1987	Internal release firewall for IBM
December 1994	IBM NetSP Secured Network Gateway V1.2
October 1995	IBM Internet Connection Secured Network Gateway V2.1
July 1996	IBM Internet Connection Secured Network Gateway V2.2
June 1997	IBM Firewall V3.1 for AIX
March 1998	IBM Firewall V3.2 for AIX and NT
March 1999	IBM Firewall V3.3 for AIX and NT

The IBM eNetwork Firewall for Windows NT's present-day cousins are the IBM Firewall for AIX, the IBM Firewall for the AS/400 and the IBM OS/390 Firewall Technologies. Across the firewall family the basic internal functions and architecture have not changed; IBM firewalls continue to fulfill the roles of both screening filter and bastion host. The IBM firewall products can be used as a toolbox to implement the functions of different firewall configurations.

The IBM eNetwork Firewall for Windows NT is built around the popular Windows NT Server 4.0 operating system from Microsoft. As such, it takes advantage of a wide array of inexpensive and widely available Intel-based PCs, giving customers a low-cost entry-level firewall product.

2.1 IBM Firewall for NT Components

At its core, the firewall is an IP gateway. Thus, it sits in between two or more networks, dividing them into trusted and untrusted zones. The firewall decides whether to pass traffic from one network to another based on connection rules. The tools used to implement the Internet security policy are:

- IP filters
- Proxy servers
- SOCKS server
- Domain name service
- Secure mail handling
- Network Address Translation (NAT)
- Virtual Private Network (VPN)

- Authentication methods
- Logging facilities
- Remote administration
- Firewall Setup Wizard

These tools and components are described in the sections that follow.

2.2 IP Filters

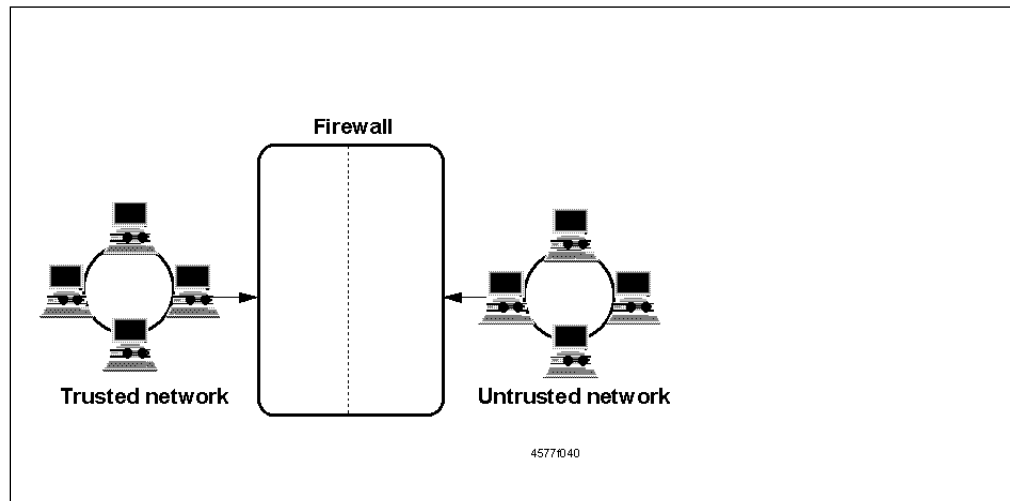


Figure 7. Firewall with IP Filtering

Filters are tools used to examine IP packets based on their source, destination, and type. The filter rules work with the IP gateway function of the firewall; as such, the firewall is required to have two or more network adapters. At least one adapter must be declared secure and another non-secure. The filters act at all of these adapters. That is, the filters not only control what can flow from the secure side to the non-secure side but also what can flow in and out of each adapter individually.

2.2.1 Objective of the Filters

IP filtering provides the basic protection mechanism for the firewall, allowing you to determine what traffic passes across it based on IP session details. This effectively protects the firewall from automated port-scanning techniques favored by crackers as an initial vulnerability probe. The IP filtering facility is the base on which the other firewall components are constructed.

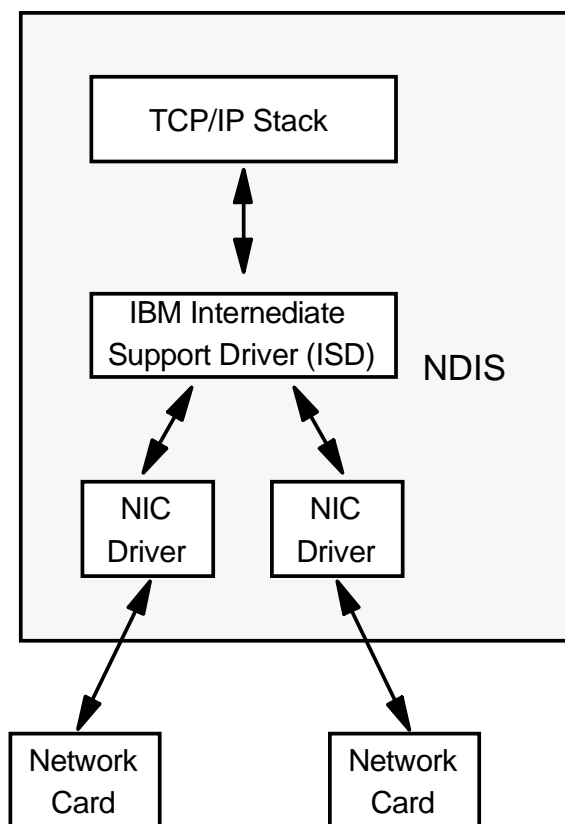
2.2.2 Implementation

The IBM Firewall intercepts IP packets based on a set of filter rules. It either discards these packets before they enter the IP transport or modifies them after they leave it. To accomplish this, the IBM Firewall installs an intermediate layer between the IP stack and the device driver for the network interface cards. Normally, the device driver binds directly to the IP stack of Windows NT.

This intermediate driver is called the IBM Intermediate Support Driver. It intercepts inbound and outbound IP traffic flowing between the native IP transport and NIC drivers bound to this transport. The intercepted traffic is routed through a frame

filter function in the controller, which can cause a packet to be discarded. The controller can also inject outbound packets.

Figure 8 shows how the IBM Intermediate Support Driver interacts with the other network components.



5209PINTISD

Figure 8. The IBM Intermediate Support Driver

The IBM Firewall implements a hierarchy of definitions for IP filters which greatly ease the task of creating and administering them. See 8.1, "Rule Base" on page 143 for more detailed information.

Table 2 illustrates this hierarchy.

<i>Table 2 (Page 1 of 2). Hierarchy of IP Filters</i>	
Level	Description
Rules	Rules are the most fundamental level for IP filters. Rules are written to permit or deny IP traffic based on source port, destination port, and packet direction. The IBM Firewall comes with a number of predefined rules, but you can also create your own using the firewall's configuration client. An example of a rule: permit initiated connection from source port greater than 1023 to destination port 80

Table 2 (Page 2 of 2). Hierarchy of IP Filters	
Level	Description
Services	<p>A service is a collection of rules used by a network application. An example of a service is HTTP, which is used for Web transactions. HTTP would be composed of the following rules:</p> <p>permit initiated connection from source port greater than 1023 to destination port 80</p> <p>permit response connection from source port 80 to destination port greater than 1023</p>
Connections	<p>A connection defines the service that is permitted or denied between two network objects. A network object can either be a host, a subnet, or a network. An example of a connection would be:</p> <p>permit HTTP from secure network to the Internet</p>

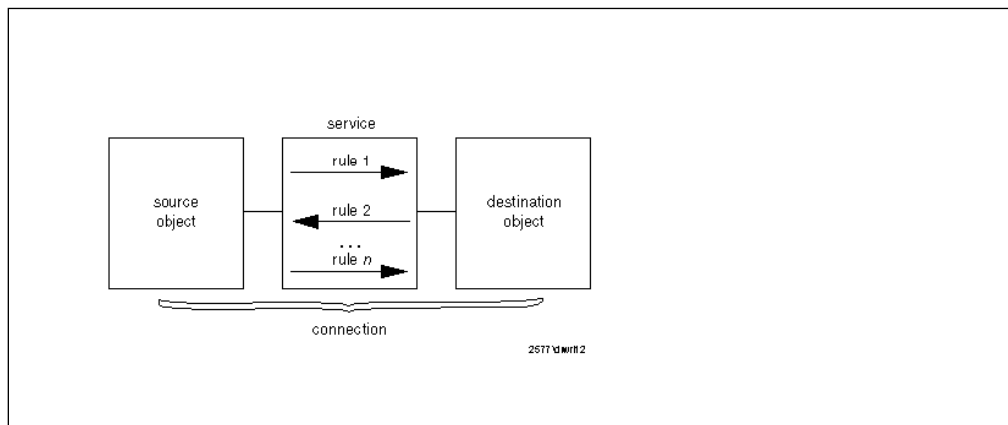


Figure 9. Rules, Services, and Connections

Thus, with the IBM eNetwork Firewall for Windows NT, you do not need to specify each of the IP filter rules. Instead, you define the connections that you want the firewall to support and the firewall generates the appropriate filter rules. This process is greatly eased by the presence of predefined services which you simply select and activate. The generated filters are defined using the following criteria:

- Source IP address and subnet mask
- Destination IP address and subnet mask
- Type of IP protocol
- Source and destination service ports, specified by TCP and UDP ports
- Direction of the IP packet
- Network interface
- Whether routed through the firewall or not
- Whether the IP packet is fragmented or not

These criteria are described in detail in Chapter 7, "Coming to Grips with IP Packets" on page 127, Chapter 8, "IBM Firewall Rule Base" on page 143, and Chapter 9, "Examples of Rules for Specific Services" on page 161. Whenever a packet arrives at the firewall, it is compared with each filter until one is found that

matches it. At that point, the processing stops and the filter directive (permit or deny) is applied. The default directive is to deny the packet.

2.3 Proxy Servers

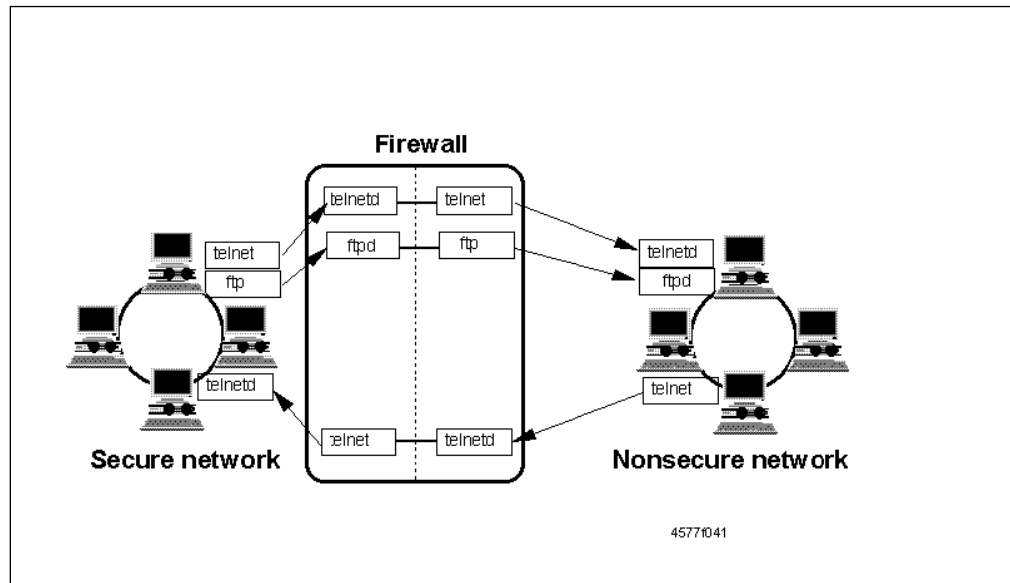


Figure 10. Firewall with Proxy Server

Proxy servers provide network security at the application level, hence, they each operate on one specific application only. A user connecting to another system through a firewall must first connect to the proxy server and ask the proxy server to provide a connection to the final destination server. Since connections are terminated at the firewall, the proxy server can examine the data of the connection before sending the data to the destination. In this way, the proxy server can provide access control based on the application data, for example, an authenticated user name. The difference between the proxy server and IP filter is that connections through proxy servers do not involve packet routing at all.

2.3.1 Objectives of the Proxy Servers

When you connect through a proxy server, the TCP/IP connections are broken at the firewall, so the potential for compromising the secure network is reduced. Compared with IP filtering, proxy servers can provide more comprehensive logging based on the application data of the connections. Similarly, users may be required to authenticate themselves using one of a number of authentication methods.

Using a proxy, there is no need to install a special version of the client program on the client machine, and once connected, the appearance and behavior of the application will remain unchanged. However, connecting via proxy implies a double-connection, first to the firewall proxy and then to the application itself; this may be time-consuming and confusing for some novice users.

A major advantage of proxy servers is address hiding. All outbound proxy connections use the firewall address. The proxy server can also minimize the impact of vulnerabilities on the client machine that would otherwise be visible over a direct connection to the destination server.

Application proxies can handle both outbound (secure network to non-secure network) and inbound (non-secure network to secure network) connections.

Inbound connections, of course, will usually require more stringent security methods.

The proxies work together with the IP filter to create a more secure firewall than one which employs either method exclusively; IP filters narrow down the list of network objects that can connect to the proxies, and the proxies provide better logging functions.

2.3.2 Implementation

The IBM Firewall includes application proxies for Telnet, FTP, and HTTP. The HTTP proxy handles not only HTTP traffic, but also other protocols (see 13.6, "Using the HTTP Proxy" on page 316). These proxy servers are installed as services on Windows NT.

The Telnet and FTP proxy servers, by default, require users to authenticate themselves at the firewall. Users accessing the gateway in this fashion will get a restricted operating environment in which only the most necessary services are provided.

The Telnet and FTP proxies can also operate in transparent mode. Under this mode, users still need to connect to the proxies but can use them without being authenticated at the firewall, so long as their connections originate from the secure network. This reduces the proxy user administration on the firewall and provides a more user-friendly interface to firewall users.

The HTTP proxy for IBM eNetwork Firewall for Windows NT complies with HTTP 1.1, and as such is capable of supporting both transparent access and proxy user authentication.

See Chapter 13, "Configuring Proxy Services and SOCKS" on page 293 for a detailed discussion of the proxies.

2.4 SOCKS Server

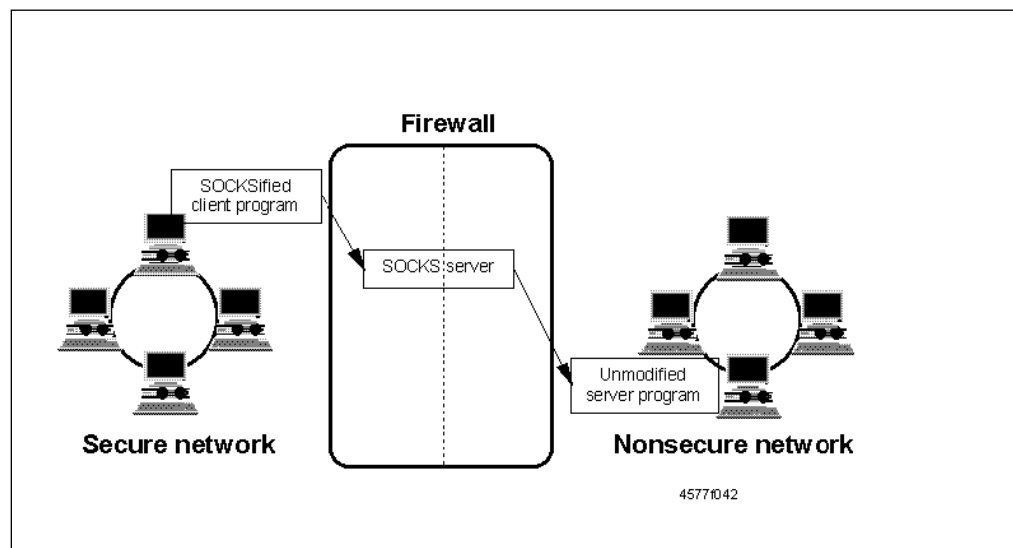


Figure 11. Firewall with SOCKS Gateway

SOCKS is a standard for circuit-level gateways. As with the other proxy types, the connection session is broken at the server: when the user starts a client application with the destination server, the client initiates a connection to the SOCKS server. The SOCKS server validates that the source address and user ID are permitted to establish the connection; from then on, the SOCKS server completes the connection.

Unlike application-level gateways, SOCKS is not application-specific. However, it does require SOCKSified clients and for older operating systems, this means a modification of the network stack. Newer operating systems now come with SOCKSified TCP/IP stacks, so SOCKS is an even more appealing option than before.

See Chapter 13, “Configuring Proxy Services and SOCKS” on page 293 for a detailed discussion on SOCKS. See also 21.2, “Aventail AutoSOCKS” on page 521 for an example of software to SOCKSify your stack.

2.4.1 Objectives of the SOCKS Server

SOCKS has the same objectives as the proxy application servers, that is, to break the session at the firewall and provide a secure gateway for access control. It also has the advantage of greater simplicity for the user, at a cost of a little extra administrative work. This is due to the modification required on the protocol stack of the client machines. However, once the SOCKSified protocol stack is installed, users can use services with no changes in their usual procedure.

With SOCKS V5 the IBM Firewall can now also process inbound authenticated SOCKS connections, not just only outbound.

2.4.2 Implementation

The SOCKS server for the IBM Firewall conforms to the SOCKS V5 specification as laid out in RFCs 1928 and 1929. As with the application proxies, it is implemented as a service on Windows NT.

SOCKS V5 supports authentication of clients using the Challenge-Response Authentication Method (CRAM) protocol. CRAM is an extension of SOCKS V5. SOCKS V5 also supports UNPW (Username/Password) authentication.

With the introduction of SOCKS V5, three types of authentication profiles are possible with the SOCKS server on the IBM eNetwork Firewall for Windows NT:

Strict All SOCKS clients must authenticate.

Permissive No authentication is necessary.

Intermediate Requires SOCKS V5 clients to authenticate but permits outbound SOCKS V4 to pass without authentication. All inbound SOCKS V4 is denied.

2.5 Domain Name Services

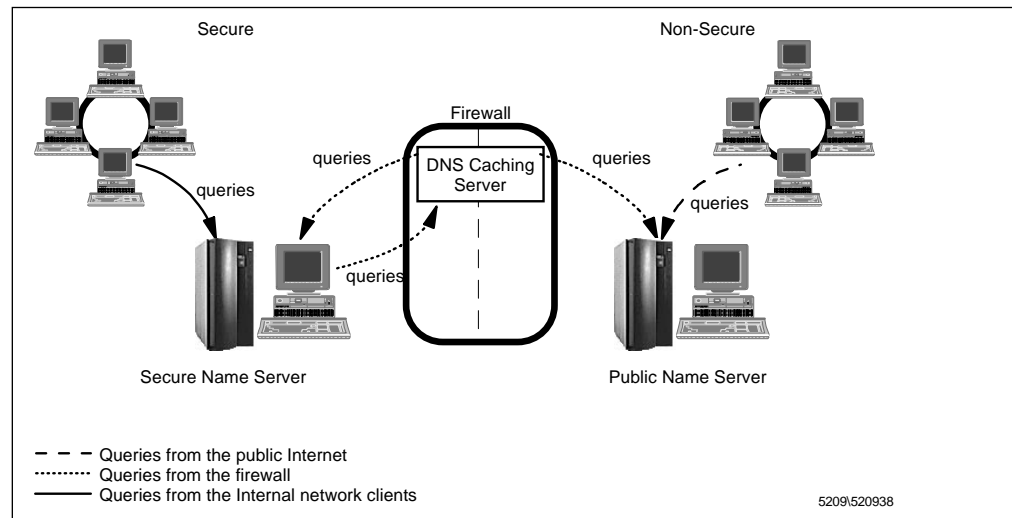


Figure 12. Domain Name Service Implementation

The firewall is able to hide the secure network's IP address from the non-secure side. This feature is important because DNS records can provide crackers with clues as to the structure of the internal network, as well as a list of potential targets. A subverted DNS server can also provide an access route for a cracker.

See Chapter 11, "Domain Name Service (DNS)" on page 247 for more information.

2.5.1 Objectives of the DNS Server

Running the DNS server on the firewall has the dual advantage of preventing name resolution requests from flowing across the gateway and hiding secure network host names from the public network.

2.5.2 Implementation

The IBM Firewall uses the Microsoft DNS Server in a cache-only mode. The Microsoft DNS Server is part of the Windows NT Server installation CD, but must specifically be installed.

The overall naming service framework for the firewall requires two additional DNS servers: a name server in the secure network, and a name server in the public network. The secure name server maintains a complete database of servers for the company's domain. The public name server only contains information about public servers of the company's domain. Internal clients use only the secure name server for resolution requests.

The secure name server must be configured to forward name requests to the firewall's DNS cache to resolve name requests not within its database. No special configuration is needed on the public DNS server.

Figure 12 illustrates this flow.

2.6 Secure Mail Handling

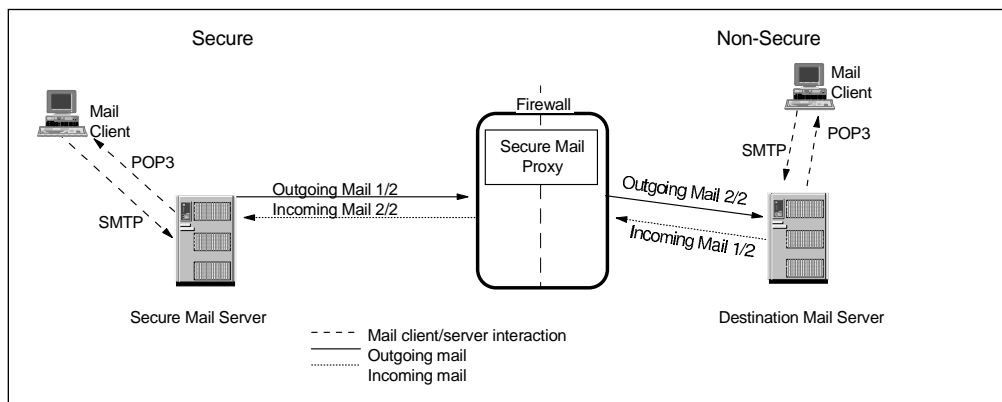


Figure 13. Secure Mail Handling Implementation

The IBM eNetwork Firewall for Windows NT uses a simple but secure mail relay called Secure Mail Proxy. It is configured to forward mail to a mail server within the secure network, without storing the mail at all on the firewall. Secure Mail Proxy also rewrites outgoing mail addresses so that details of the internal network can remain hidden.

2.6.1 Objectives of the Mail Relay

As with the proxy services, Secure Mail Proxy provides a means to break the connection session in two, thus protecting the secure mail server. It also hides the internal mail gateway from the public network. Only the firewall is advertised outside the secure network.

2.6.2 Implementation

Since Secure Mail Proxy is a mail relay and not a mail server, there must be mail server(s) in the secure network to receive and store all incoming mail. Going in the opposite direction, the secure mail server(s) must forward all outgoing messages to Secure Mail Proxy for relay to their final destination.

From the point of view of the hosts in the non-secure network, the only mail gateway they see is the firewall.

Figure 13 illustrates this flow.

2.7 Network Address Translation (NAT)

With the incredible growth of the Internet over the past few years it is getting difficult to get as many unique IP addresses as there are hosts connected to it. The best solution is to use reserved IP addresses for Intranets. But how do you communicate to the Internet? Proxies and SOCKS Servers are one way to do this but when it comes to routed IP traffic you have to choose another solution.

2.7.1 Objectives of Network Address Translation (NAT)

NAT can do three things for you:

1. Allows you to communicate with the Internet by using private addresses
2. Hides your internal network structures
3. Routes traffic to normally non-routable IP addresses

Though NAT looks like a good choice, you have to remember that you have a direct connection to the Internet. There is no breaking of connections like in application level gateways. NAT comes in handy if you have some protocol that is not supported by any proxy or SOCKS server.

2.7.2 Implementation

Basically NAT just describes the process of translating one (internal) IP address to another (external) IP address. The implementation of the IBM eNetwork Firewall for Windows NT is a little different from the AIX version. NAT in AIX is the so-called traditional NAT and translates internal IP by allocating an external IP out of a pool of reserved IPs. This pool IP will be given back when it is no longer used. On the NT version only one IP address is used but also the ports are allocated so you get a lot of IP/port pairs (more than 64,000) instead of just some IP addresses. This kind of translation is called Network and Address Port Translation (NAPT).

2.8 Virtual Private Network

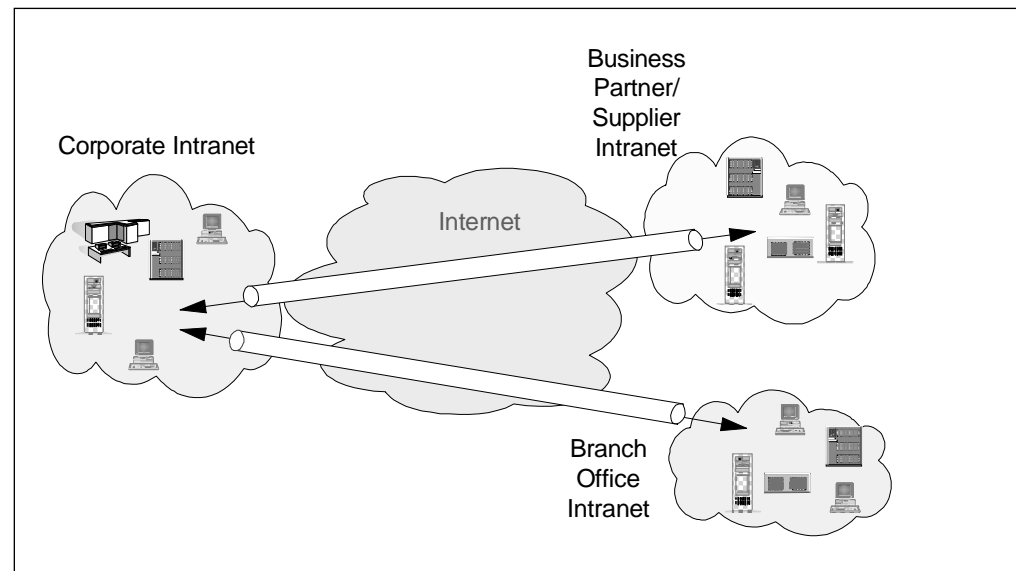


Figure 14. Virtual Private Network Implementation

The IBM eNetwork Firewall for Windows NT supports the connection of two secure networks to extend the private network of an enterprise, over a backbone network, such as the Internet. A virtual private network (VPN) will be created by configuring a tunnel between the two secure networks. Once the VPN is established, it's like being in one secure network completely transparent to the users.

2.8.1 Objectives of a Virtual Private Network

A VPN creates a secure connection that protects your data while being transferred from one secure network over the nonsecure network to the other secure network. To protect the data the VPN tunnel supports the IPSec security standards, which provide authentication of the datagrams, encrypt and validate data and establish and maintain cryptographic keys so the data is kept private and is hidden from eavesdroppers on the nonsecure network.

2.8.2 Implementation

To establish a tunnel with the IBM eNetwork Firewall for Windows NT, the partner end must support the new VPN headers (see 15.1, “Secure IP Tunnel Standards - Interoperability” on page 357). The IBM eNetwork Firewall for Windows NT can establish tunnels with another IBM eNetwork Firewall for Windows NT, OS/390 Firewall Technologies V2R6, AIX 4.3, AS/400 Operating System and 2210/2216 routers.

Note: The IBM eNetwork Firewall for Windows NT cannot establish a tunnel with the IBM eNetwork Firewall for AIX 3.3.

You can manually configure the VPN tunnels using the Firewall GUI. The concept of a tunnel carrying encrypted or authenticated data or both is integrated in the IP filtering rules. These filtering rules can be either dynamic or static.

Dynamic filter rules are implicitly activated when a VPN tunnel is activated.

Static filter rules are manually configured and must be explicitly activated to take effect. With static filters you have the ability to define filter rules for each specific application.

See Chapter 15, “Virtual Private Network” on page 357 for more information on the implementation.

2.9 Authentication Methods

The firewall supports a variety of authentication methods, ranging from simple passwords to one-time passwords as implemented by security tokens. These authentication methods are a must for firewall administration. These can also be used to control access to SOCKS and application proxy services.

2.9.1 Objectives of the Authentication Methods

Authentication methods are a definite requirement for verifying administrators who will be configuring the firewall. They may also be used for verifying access to SOCKS and proxy services, if the security policy so requires.

Simple passwords are a weak point of security as they may be easily guessed or compromised due to carelessness on the users' part. One-time passwords and tokens enhance the security of the firewall by practically eliminating password-guessing attacks.

Authentication methods, especially the stronger ones, are of great practical use when dealing with inbound connections from the non-secure network.

2.9.2 Implementation

The firewall implements two types of simple passwords: a firewall logon password and an NT logon password. The firewall logon password for both administrators and users is supplied from the firewall configuration client and stored in an MD5-encoded hashed password file on the system. The NT logon password, on the other hand, synchronizes the administrator and user passwords for firewall services with the Windows NT account password. The Windows NT account password may either be supplied to the specific local machine or the Windows NT domain.

The firewall also supports one-time passwords using tokens such as SecurID from Security Dynamics. These tokens generate non-repeating sequences of numbers which are checked against an authentication server in the secure network. The firewall communicates with the authentication server to verify logons.

The firewall provides an API for customized authentication methods. These are documented in the *IBM eNetwork Firewall for Windows NT Reference Version 3 Release 3* manual.

See Chapter 13, “Configuring Proxy Services and SOCKS” on page 293 for more information on the implementation.

2.10 Logging

Logging is essential to the day-to-day operation of a firewall. Assiduous examination of the logs, whether done automatically by the firewall or through computer-assisted analysis by the administrator, can reveal suspicious activities which have taken place or are still in progress.

The IBM Firewall comes with extensive logging capabilities which can record events relating to the IP filters, proxies, SOCKS, mail, and authentication.

See Chapter 16, “Logging, Monitoring, and Reporting” on page 409 for a discussion of the firewall's logging functions and how to use them effectively.

2.10.1 Objectives of Logging

Logging provides a facility in which practically all events relating to the firewall can be recorded. These logs can be used to trigger an alert to administrators when suspicious activities occur. Used over a longer period as historical records for analysis, the logs can show if a deliberate attack taking place at different intervals was being attempted from some source.

Logs can also be used for accounting purposes to show the usage of network services by users.

2.10.2 Implementation

Each event on the firewall, from severe errors down to the most innocuous activity, generates a specific message. These messages are known as ICA messages, so called because of the first three fixed bytes of each message tag.

There are seven categories of ICA messages, each category relating to facility of the firewall:

- 0000-0999 Intrusion alarm
- 1000-1999 Filters
- 2000-2999 Proxy
- 3000-3999 SOCKS
- 4000-4999 Pager
- 5000-5999 Secure Sockets Layer
- 6000-6999 Virtual Private Network
- 7000-8999 Available for future use
- 9000-9999 General/others

Each message falls into one of four severity levels. In order of increasing severity, these are:

1. Information
2. Warning
3. Error
4. Severe

These messages can be written to a log file specified by the administrator. The administrator may choose to record only a specific severity level and those above it. There is also a debug mode which records all messages regardless of severity.

The firewall can be configured to take a course of action if a particular event or sequence of events, as indicated by the logged messages, is triggered. Such an action may involve notifying the administrator by e-mail or pager, or executing a program or script.

Archiving utilities are included with the firewall. Run at regular intervals, these utilities will keep the logs to a manageable size.

Also included with IBM eNetwork Firewall for Windows NT are DB2 scripts for converting these files into DB2-readable format. An API is also included for integrating the firewall with other logging programs.

See Chapter 16, "Logging, Monitoring, and Reporting" on page 409 for more information on the implementation.

2.11 Remote Administration



Figure 15. The Firewall Configuration Client

The IBM Firewall's configuration client can be installed on a Windows NT or Windows 95 machine, and the task of maintaining the firewall can be delegated to other administrators. The communication between the configuration client and the firewall itself may also be encrypted using SSL. This is especially useful if the firewall must be configured from the non-secure network.

2.11.1 Objectives of Remote Administration

Firewall administration delegation is an option companies might want to consider if they need more than one administrator to maintain the firewall. For example, a backup administrator may be appointed to reconfigure some portions of the firewall when the primary administrator is unavailable.

Remote configuration is useful in large networked environments where a knowledgeable administrator may not be in the local vicinity of the physical machine. It is also useful for delegating firewall tasks to secondary administrators, as will be discussed in Chapter 10, "Remote Administration Configuration" on page 221.

2.11.2 Implementation

The configuration client is a separate component that can be installed on another Windows NT, Windows 95, or AIX machine. By default, it communicates with the firewall server on port 1014.

The primary firewall administrator (the account which was used to install the firewall) can manage all aspects of the firewall, as well as create other firewall administrators. These secondary firewall administrators can only manage specific portions of the firewall as delegated to them by the primary administrator; they cannot create other administrator accounts.

The firewall includes a utility for creating a self-signed SSL key, which is then used to encrypt the communication between the firewall and the configuration client.

2.12 Firewall Setup Wizard

A wizard has been provided to aid the user with the initial configuration of the IBM eNetwork Firewall for Windows NT. This setup wizard enables a user, who does not have extensive knowledge of the firewall, to have a basic firewall configuration up and running quickly after installation. The wizard guides you through the necessary steps configuring interfaces, DNS, mail proxy, log setup and Web access.

2.12.1 Objectives of Setup Wizard

Running the wizard enables you to reduce the initial configuration set-up time and to curb configuration usability problems.

2.12.2 Implementation

The usage of the wizard is optional. It comes up automatically the first time, afterwards it is available from the help pull-down menu.

- Wizard is tied in with GUI, runs wherever GUI runs
- For this release the Wizard will be included with for Windows NT only. For AIX it will be available for the next Firewall for AIX release.
- Anything the wizard does can be undone by using the configuration client.
- It is easy to build on the configuration that the wizard has done.

For some situations, it is not recommended to use the wizard. For example:

- Migrating from a previous release
- Multiple network adapters on the secure side of the firewall for which there is more than one security policy.

2.13 From Here On...

This chapter has given you an overview of the IBM eNetwork Firewall for Windows NT, discussing each of its basic components. For more detailed information, you may want to refer to the following chapters:

- To learn more about IP filters as implemented in the IBM Firewall, see Chapter 7, "Coming to Grips with IP Packets" on page 127, Chapter 8, "IBM

Firewall Rule Base” on page 143 and Chapter 9, “Examples of Rules for Specific Services” on page 161.

- To learn more about proxies, SOCKS, and authentication methods, see Chapter 13, “Configuring Proxy Services and SOCKS” on page 293.
- To learn more about the implementation of DNS, see Chapter 11, “Domain Name Service (DNS)” on page 247.
- To learn more about secure mail handling, see Chapter 12, “Mail Handling” on page 269.
- To learn more about Network Address Translation see Chapter 14, “Network Address Translation” on page 339.
- To learn more about Virtual Private Networks see Chapter 15, “Virtual Private Network” on page 357.
- To learn more about logging, alerts, and reports, see Chapter 16, “Logging, Monitoring, and Reporting” on page 409.
- To learn more about remote administration, see Chapter 10, “Remote Administration Configuration” on page 221.
- To learn how to install the IBM eNetwork Firewall for Windows NT, see Chapter 4, “Installing the IBM eNetwork Firewall for Windows NT” on page 43.
- To learn how to configure IBM eNetwork Firewall for Windows NT, see Chapter 5, “A Quick and Basic Configuration: A Case Study” on page 77.
- To learn how to configure IBM eNetwork Firewall for Windows NT with the new wizard, see Chapter 6, “Firewall Setup Using the Wizard” on page 109.

Chapter 3. Planning Your Firewall Installation

Before you rush off to install the IBM eNetwork Firewall for Windows NT, you would do well to step back first and decide how you want to implement your security policies. This chapter contains the general principles for network security within the framework and capabilities of a firewall. It is intended to be a checklist of steps and decision points you will want to consider in rolling out your firewall.

3.1 Defining a Firewall Security Policy

A firewall is neither the beginning nor the end of network security. It is simply a tool for enforcing a company's policies with regard to networks and the Internet. As such, a firewall is only effective if its role in protecting the enterprise is well-defined. Network security has its foundations not so much in technology as it does in the company's security policy.

A security policy is, in effect, a mission statement. With the security policy, you define what things are of importance to the company and, if needed, assign various degrees of sensitivity to each. You also define who your users are, if possible, grouping them according to the level of access and types of activities which you will permit them.

A security policy also serves as a reality check. After examining your resources and your users, you should now be able to gauge more carefully what security measures are appropriate. Your security should be commensurate to the value of the object you are protecting. Similarly, your security should not be so severe as to impede legitimate users.

Every organization should have a corporate security policy, a policy put forward with the blessing, support, and authority of upper-level management. The corporate security policy should be known and understood by all, as should the corresponding sanctions for violations. Furthermore, the policy should also discuss what the contingencies are in the case when security is violated.

The corporate security policy determines your network security policy. The network security policy in turn determines your firewall security policy. The same principles apply across these three levels with minor modifications to accommodate the nuances of each.

3.1.1 Defining Resources

"What do you want to protect?" is the simple question that, answered properly, reveals the resources that form your security objects. You will certainly want to protect corporate information, of course, and conversely, the systems on which the corporate information resides. In many cases, the answers may not be so clear cut; different types of information will have different degrees of value, and would therefore merit different levels of security.

For firewalls, these resources are invariably computational or data servers. With regard to firewall security planning, therefore, you should make a list of these servers, the data they contain, and the users who will be accessing them.

For the implementation of resources definition on the IBM Firewall for NT, refer to Chapter 8, "IBM Firewall Rule Base" on page 143.

3.1.2 Defining Users

Who will be using these resources? The answer to this question will be a list of different groups of users who will have access to the resources you have listed. These different groups will have different privileges to your servers. Take note: users are not only your company's employees but may also include business partners and, in the case of public Web servers, the world in general.

With respect to firewalls, users will invariably be anyone who has a possible access to your servers, whether directly to the system itself or through the network. In the case of the Internet-connected networks, this could refer to anyone in the world! Thus the firewall becomes a dividing point between the secure and the non-secure networks; in more complicated scenarios, it also covers the various shades between the two.

For the implementation of user definition in the IBM Firewall for NT, refer to 13.1, "User Administration" on page 294.

Subnets

Large networks with numerous servers may be divided into smaller subnets, reflecting the organizational structure of the company. Identifying these subnets to the firewall can be useful in grouping together the different types of users and resources; where applicable, blanket policies may be put into effect. Apart from the administrative benefits, this also improves performance by reducing the number of definitions of users, network objects, and rules on the firewall itself.

3.1.3 Defining Activities

Having defined resources and user groups, the next exercise will be to define the permitted interaction between them. Put in another way: What activities are you willing to permit on your servers and on your network? With regard to the firewall, these activities may be categorized into outbound services and inbound services. Outbound services are activities that you will permit from the secure network to the non-secure network. Inbound services are activities which go in the opposite direction: from the non-secure network to the secure network.

With regard to the firewall, there is good news and bad news. The good news is that services are defined in terms of protocols and TCP/IP port numbers which are reasonably well-defined. The bad news is that the task of defining these protocols on the firewall becomes unwieldy and cumbersome the more services are permitted. In this light, the KISS principle -- "Keep It Simple and Secure" (for the less vulgar expansion) -- is especially appropriate.

For the definition of activities in the IBM eNetwork Firewall for Windows NT, refer to Chapter 8, "IBM Firewall Rule Base" on page 143, Chapter 13, "Configuring Proxy Services and SOCKS" on page 293, and Chapter 17, "Configuring the IBM Firewall for Three Adapters" on page 449. For examples, see Chapter 9, "Examples of Rules for Specific Services" on page 161.

The Two Security Stances

There are two stances to firewall security and to security policies in general: the permissive, which says "That which is not prohibited is permitted;" and the prohibitive, which says "That which is not permitted is prohibited." For firewalls, the correct approach is to take the prohibitive approach: begin with a fully closed system and open up services only when and as you need them. IBM eNetwork Firewall for Windows NT supports this model.

3.1.4 Administration

Who will maintain and enforce the security policy? A security policy must be backed up by monitoring mechanisms and authority to action. This authority to action must in turn be strongly supported by upper management (a requirement sadly neglected in some implementations.)

The task of managing a firewall falls to the administrators who look after it on a day-to-day basis. The security policy must therefore identify the administrators who will maintain the firewall. In the case of large organizations, there might even be multiple levels of administration, each delegated to perform a distinct task on the firewall.

For the implementation of administration on the IBM Firewall for NT, refer to Chapter 10, "Remote Administration Configuration" on page 221.

Another redbook, *The Technical Side of Being an Internet Service Provider*, SG24-2133 also devotes a section to developing Internet security policies. Refer to it for more information.

3.2 Defining Network Objects

The resources you've defined in your security policy translate to network objects when viewed from the perspective of the firewall. These network objects are computer hosts and servers, identified by the firewall by their IP addresses. When configuring a VPN, we also need to define the remote hosts, the remote secure network and the partner tunnel endpoint.

If you have a large number of computers in your organization, careful planning of your network will permit you to group servers with common security restrictions into their own subnets within the intranet. This is ideal from a firewall perspective, because you reduce the number of rules that you will have to enforce. The firewall will be more efficient, and you simplify the task of administration.

Not only compute or data servers can be identified as network objects. Where possible, client computers should also be defined. Placing client computers within a common subnet is almost a prerequisite, given that their number could range from anywhere in the tens or hundreds or even more. However, clients with special access to the firewall or through the firewall should be specifically identified. An example of this would be the firewall's remote administration clients or remote clients using a tunnel connection.

Sample tables for planning your network objects are shown below. We use Figure 16 on page 32, also the same network in the example given in Chapter 5,

“A Quick and Basic Configuration: A Case Study” on page 77, as the basis for our examples.

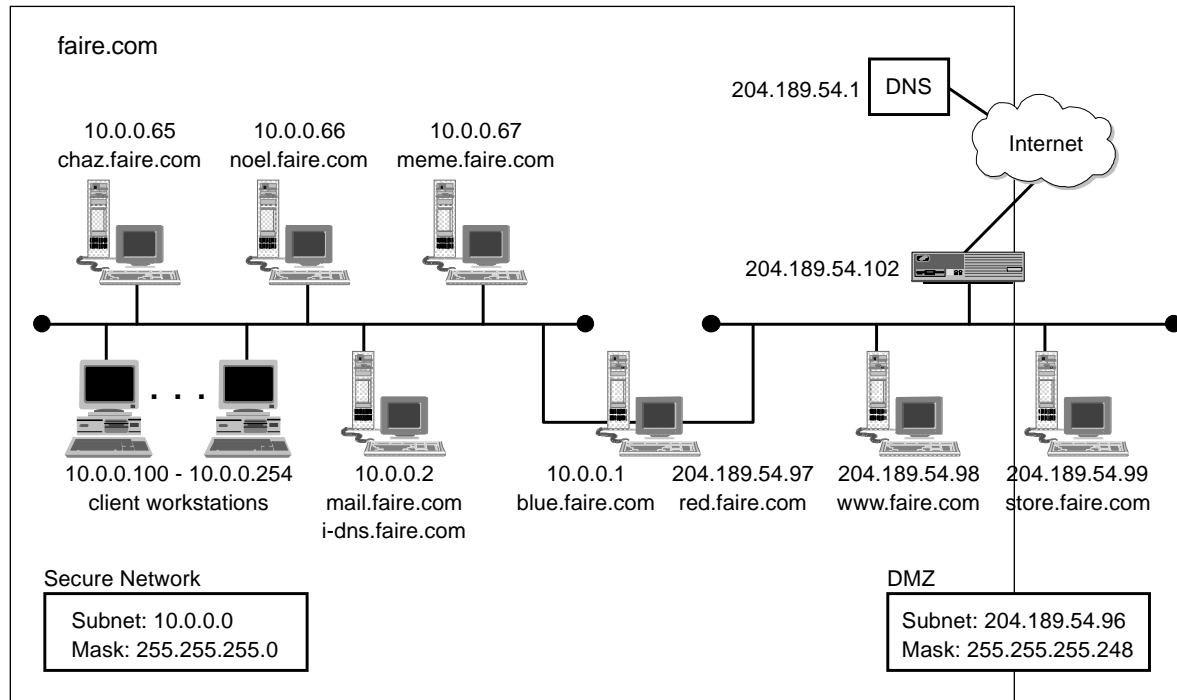


Figure 16. A Sample Network

Table 3 is a table which defines the networks in our domain of concern.

Network Name	Network Number	Mask	Comments
Secure Network	10.0.0.0	255.255.255.0	Private network using RFC 1918 addresses
DMZ	204.189.54.96	255.255.255.248	Subnet provided by ISP
ISP Network	204.189.54.0	255.255.255.0	Shared network provided by ISP, non-secure!
The World	0.0.0.0	0.0.0.0	Everybody else. Trust no one.

Table 4 identifies the servers and other objects in the network.

Host Name	IP Address	Services Provided	Type	Network
chaz.faire.com	10.0.0.65	Intranet Web service (80/tcp) Intranet ftp service (20/tcp, 21/tcp)	Windows NT server	Secure Network
meme.faire.com	10.0.0.66	DB2 Universal Database (1501/tcp, 1502/tcp)	Windows NT server	Secure Network
noel.faire.com	10.0.0.67	Lotus Notes/Domino (1352/tcp)	Windows NT server	Secure Network

Host Name	IP Address	Services Provided	Type	Network
mail.faire.com	10.0.0.2	Lotus Notes/Domino (1352/tcp)	Windows NT server	Secure Network
		SMTP mail service (25/tcp)		
		POP3 mail service (110/tcp)		
		DNS (53/tcp, 53/udp)		
blue.faire.com	10.0.0.1	Filter, proxy services	Secure firewall adapter	Secure
red.faire.com	204.189.54.97	Filter, proxy services	Non-secure firewall adapter	DMZ
www.faire.com	204.189.54.98	Public Web server (80/tcp)	Windows NT server	DMZ
store.faire.com	204.189.54.99	Public Web server (80/tcp)	Windows NT server	DMZ
		Secure Web server (443/tcp)		
internet_router	204.189.54.102	-	IBM 2210 router	DMZ/ISP
dns1.isp.com	204.189.54.1	DNS (53/tcp, 53/udp)	unknown	ISP

Table 5 defines clients who will be accessing resources in the network. (The definition of the remote administration client was added as a sample entry. It is not reflected in the reference diagram.)

Host Name	IP Address	Comment	Network
Secure client workstations	10.0.0.101 to 10.0.0.254	Windows NT Workstation/Windows 95	Secure
admin.faire.com	10.0.0.100	Windows NT, remote administration client	Secure
The World	0.0.0.0	Web users connecting to public servers (the huddled unwashed masses)	The World

3.3 Defining Connections

Having defined the network objects, the next logical step would be to define the connections between them. Define only the barest minimum number of connections to enable the services you want. These connections will involve the services your servers will provide, and the clients which will connect to these servers.

Determine the direction of the traffic from the source network and the destination network. If the traffic goes through the firewall, consider also how the connection will take place. IP traffic are the most basic connections you will configure through the filter rules, but consider using proxies (see 3.4, "Proxy Decisions" on page 34) or other firewall services where applicable.

A sample table for defining connections is given in Table 6 on page 34. Not all the connections will involve the firewall, but all are listed for the sake of completeness.

Table 6. Defining Connections

Server	Service/Port Number	Client	Network Source/Destination	Connection Type
chaz.faire.com	HTTP/80	Secure client workstations	Secure/Secure	Direct
	FTP/20, 21, unreserved	Secure client workstations	Secure/Secure	Direct
meme.faire.com	DB2/1501, 1502	Secure client workstations	Secure/Secure	Direct
noel.faire.com	Domino/1352	Secure client workstations	Secure/Secure	Direct
mail.faire.com	Domino/1352	noel.faire.com	Secure/Secure	Direct
	SMTP/25	Secure client workstations	Secure/Secure	Direct
	SMTP/25	blue.faire.com	Secure/Secure	via SafeMail
	POP3/110	Secure client workstations	Secure/Secure	Direct
	DNS/53	Secure client workstations	Secure/Secure	Direct
	DNS/53	blue.faire.com	Secure/Secure	Direct
	POP3/110	Secure client workstations	Secure/Secure	Direct
www.faire.com	HTTP/80	Secure client workstations	Secure/DMZ	SOCKS
	HTTP/80	The World	The World/DMZ	Direct
store.faire.com	HTTPS/443	The World	The World/DMZ	Direct
dns1.isp.com	DNS/53	red.faire.com	DMZ/ISP	Direct
	DNS/53	The World	The World/ISP	Direct
The World	SMTP/25	red.faire.com	DMZ/The World	via SafeMail
	HTTP/80	Secure client workstations	Secure/The World	SOCKS
	FTP/20, 21, unreserved	Secure client workstations	Secure/The World	SOCKS

See Chapter 8, "IBM Firewall Rule Base" on page 143 and Chapter 9, "Examples of Rules for Specific Services" on page 161 for more information.

3.4 Proxy Decisions

The proxy services, working in concert with the IP filters on the IBM eNetwork Firewall for Windows NT provide a more secure way to manage network connections going through the firewall than IP filter rules alone. Proxy services can enforce passwords, strong authentication, and other decision issues that are closely tied in to the network applications themselves.

IBM eNetwork Firewall for Windows NT provides proxy servers for Telnet, FTP, and HTTP as well as the general-purpose SOCKS circuit-level gateway. (The HTTP

proxy can also process WAIS, Gopher, and HTTPS requests.) These proxies provide you with greater granularity of control to the services you are going to provide for your users. Among the decisions you should make with proxies:

- Which type of proxy to use: application-level proxies or SOCKS. SOCKS is a simple general-purpose gateway that can relay several protocols, applying a common set of policies to all of them. It's completely transparent to users but requires SOCKSified clients. Application proxies, on the other hand, are specific to one application each, and so permit fine-grained control. They do not require modification of the clients, but they are not completely transparent to users.
- Whether users need to authenticate themselves before being allowed to use the proxy servers. If users do not need to authenticate, the application proxies can operate in transparent mode. Otherwise, they can use a wide variety of methods: simple passwords, strong authentication via one-time passwords, NT domain logon synchronization, or any custom-written authentication scheme.
- If you use simple passwords for authentication, what type of password policy should be enforced. IBM eNetwork Firewall for Windows NT's password policy allows you to set the characteristics of the passwords, such as minimum length, number of repeating characters, history lists, minimum time before change, etc.
- Whether to permit access to the secure network from the non-secure side. Both SOCKS and the application proxies provide a safe way to implement this. In both cases, you should strictly define who can access what resources in the secure network at what times and from which computer or network they can do so.

Once you've decided which applications you will support with your proxies, you should then set up IP filter rules to limit traffic to your your firewall to only those applications.

A sample table for defining users or user groups is given in Table 7. In this table, you can also fill in the proxying method they will use as well as other parameters that can be controlled or are related to the proxy service. You may want to expand this table to include specific user names in groups.

User/Group	Network/Host	Services	Destination Server	Proxy	Authentication Method
employees	Secure Network	HTTP	The World	SOCKS	none
employees	Secure Network	FTP	The World	SOCKS	none

See Chapter 13, "Configuring Proxy Services and SOCKS" on page 293 for more information.

3.5 Network Address Translation Decisions

Comparing SOCKS and NAT to the user's point of view is very similar in service because in both cases you have a transparent connection that handles most TCP and UDP protocols. The important thing to keep in mind is, that SOCKS breaks the connection at the firewall and NAT does not. Actually when you are not using NAT or VPN or any routed traffic through the firewall, you don't need the Windows NT IP Forwarding facility.

- Using NAT is more or less like having a normal routed connection to the Internet, except that your internal network cannot be seen from the nonsecure network.
- When using NAT you should consider using it carefully and only in those cases where other proxies or SOCKS will fail.
- If you don't have a DMZ, a NAT mapping to an internal server could help you to protect a server instead of having it connected directly to the Internet.

3.6 Naming Services

The firewall and the network it is protecting will undoubtedly form part of a larger network, and large networks on the whole require a name service which uniquely identifies the participating computers. In the case of the Internet, this would be the Domain Name Service (DNS).

Firewall security extends to naming services by providing a means with which to hide the names of the servers in the secure network. This configuration is called *hidden DNS* or *split-brain DNS*. In a hidden DNS configuration, there is an announced name server which contains a minimal list of publicly accessible hosts and servers; and there is a restricted name server which actually contains the real information about the secure network. The rationale is that knowledge of server names will divulge to malicious hackers the structure of the internal network and, correspondingly, possible targets for attack.

There are two possible configurations for hidden DNS: *same-name hidden DNS*, and *different-name hidden DNS*. With same-name hidden DNS, the secure and non-secure networks use the same domain name, while the opposite is true of different-name hidden DNS. It must be emphasized that in both cases, the DNS zones are distinct. That is, the non-secure network maintains a different name database from the secure network.

Overall, different-name hidden DNS is slightly more complicated than same-name hidden DNS. The two domain names must be registered with the InterNIC to preclude any possible conflicts with another company using the same domain name. The mail infrastructure must also be configured to receive mail for the two domain names.

A Third Hidden DNS Option?

What if you are using a subdomain of your domain name for your secure network? Let's say that your domain is `faire.com`, so for your internal network, you use `secure.faire.com`. Isn't this another option for DNS? Not exactly. This is a special case of same-name DNS. In your secure network, you must set up a zone for `faire.com` and from it define the subdomain, `secure.faire.com`.

Hidden DNS is not mandatory with firewalls. In some special cases, companies may have reasons for revealing internal host names to the rest of the world. In cases like this, the *open DNS* configuration is appropriate.

For firewall planning and implementation, you must decide which DNS option you are going to use and what the domain names of your secure and non-secure networks will be. If your network is going to be connected to the Internet, then you must also register your domain name with the InterNIC (see <http://rs.internic.net> or your local domain registrar for more information).

For details of implementation of DNS for the firewall, see Chapter 11, "Domain Name Service (DNS)" on page 247.

3.7 Mail Services

Mail is a sensitive service. It is the least common denominator for services on the Internet. At the same time, it has historically had more than its fair share of woes as far as security concerns go. The firewall aims to alleviate the security problems with mail by providing a secure relay between the organization's mail servers and the Internet at large.

For firewall planning and implementation, you must identify to the firewall your domain name and the Internet mail servers you will be using. The firewall will only accept mail destined for your domain name, after which it will forward them to the appropriate mail servers. For details, see Chapter 12, "Mail Handling" on page 269.

3.8 Virtual Private Network

A virtual private network is an extension of our own private network, which means we transfer data over a nonsecure backbone to a secure network. This means we have to take a few security aspects into account.

- Do we need to connect one host to one remote host, or do we need to connect our secure network to the remote secure network or to one remote host?
- Which users may have access, or do all the users need to have access?
- Are users from the remote network allowed to have access to our secure network?
- When creating the tunnel, does the partner supports the IPSec standards with the new VPN headers?
- Is the data we are transferring confidential?

- Does the data need to be encrypted, or do we need only authentication, or do we need both?
- What encryption will we use?
- Does the partner host supports this method?
- Will we use dynamic or static tunneling?
- Will all the traffic pass through the tunnel or are only some applications allowed?
- Will the tunnel be open constantly, or only for a certain period of time of the day?

Before installing the tunnel we must take all these considerations into account to get the configuration that is best for us.

3.9 Alerts, Reports, and Logs

A good firewall logs network traffic directed to it. These records can reveal much about possible intrusion attempts (whether in the past or in progress) and firewall performance issues. These firewall logs can be used in two ways: *alerts* and *reports*.

Alerts refer to actions that the firewall takes when certain events are triggered. The firewall can automatically keep track of events that you specify. These events, whether taking place once, taken together, happening in sequence, or occurring repeatedly over a period of time, will flag a significant event to which you must be alerted.

Reports and logs provide a facility for you to analyze the events that the firewall has recorded. Alerts are fine as far as warning you of possible attacks taking place; however, some security breach attempts are more insidious, taking place over a long period of time or taking advantage of a newly discovered weakness. For cases like this, you cannot rely on simple trigger rules that you tell the firewall to watch out for; you must go through the logs to determine any anomalies in the firewall's usual activities.

Accounting information is a side-benefit of the logging and archiving function. By examining the logs, you can generate records of the usage of the network, in particular, of traffic that goes through the firewall. A sample application would be variable billing of departments where the monthly Internet access charges are shared by different groups within a company.

Alerts can be built by telling the IBM eNetwork Firewall for Windows NT to watch out for the messages it generates. A complete list of these messages can be found in the *IBM eNetwork Firewall for Windows NT Reference Version 3 Release 3*.

Archives of the messages generated by the firewall must be managed as well. You will want to specify in your firewall security policy how often you will examine the consolidated logs and when to remove them from the firewall. Appendix C, "Recommended ICA Message Tags" on page 607 discusses logging, alerts, and archiving. It also provides a comprehensive list of alerts to tell the firewall to watch out for.

3.10 Administration and Firewall Management

As discussed in 3.1.4, "Administration" on page 31, administrators must be assigned to manage the firewall. There must be a primary firewall administrator in charge of maintaining all aspects of the firewall. At the same time, it makes good sense to assign other administrators to certain tasks of the firewall, in case the primary firewall administrator is unavailable.

With the IBM Firewall, delegation of tasks means the introduction of remote administration, using the firewall's configuration client (see Chapter 10, "Remote Administration Configuration" on page 221.) Of course, there are other reasons why remote administration would be desirable. In any case, the issue does bring open up other decision points for consideration:

- Who the administrators will be, and what accounts they will be using.
- What administrative privileges each admin account will have.
- Which specific client machines will be permitted to open a remote configuration connection to the firewall. This is a very important decision to make, and must be properly reflected on the firewall. It is unsafe and unwise to permit just any machine to make the connection. The remote configuration clients must also be made secure.
- What authentication schemes the administrators will use to log on to the firewall. Simple passwords maintained at the firewall might suffice, but strong authentication using one-time passwords and security tokens are a wiser choice. Whatever the selection, a corresponding password policy, more stringent than that for proxy users, must be created.
- Whether remote administration will be permitted from non-secure networks, and if so, how to implement it. At first glance, this may not be such a wise option, but some circumstances will call for it. As before, the client must be identified by IP address, if possible. Strong authentication and encryption schemes must also be supported.

It's also a good idea to maintain a journal of when the administrators logged on to the firewall. By comparing the paper audit trail with the firewall system's logon records, unauthorized access to the machine can be determined.

Table 8 shows a planning table for defining administrator rights. The rights are based on the selectable options from the firewall's configuration client.

Administrator	Network Objects	Traffic Control	Modify DNS	Modify Mail	Proxy Admin	User Admin	Log Facilities	Log Monitor	Pager	Interface	Login From
Primary Firewall Administrator	x	x	x	x	x	x	x	x	x	x	local, admin.faire.com
User Administrator					x	x					admin.faire.com

Table 8 (Page 2 of 2). Defining Administrator Rights

Administrator	Network Objects	Traffic Control	Modify DNS	Modify Mail	Proxy Admin	User Admin	Log Facilities	Log Monitor	Pager	Interface	Login From
Log Administrator							x	x	x		admin.faire.com

3.11 Mars Attacks!

Then comes the Moment of Truth: crackers are attempting to break into your corporate intranet. How do you determine if an attack is in progress?

Now, an attack does not necessarily mean a breach in firewall security. What it is is a deliberate attempt to probe your defenses. Attacks have certain characteristics that you can watch out for.

One characteristic of cracker attacks in progress is unusually high packet activity from a site or group of sites, a sign that they are probing for weaknesses in the firewall. This is immediately noticeable because the firewall will log these packets like mad, thus resulting in a slowdown of firewall performance. By setting alerts, administrators can immediately be notified of these ongoing events.

Another indicator of a possible attack is the shutdown of some of the firewall's services. This could mean that the services have crashed from servicing too many requests, or that they are being restarted to take in new configuration parameters.

But not all attacks are so obvious. Slow, deliberate attacks taking place at distant intervals are quite possible as well, and not so easily detectable. Worse still are attacks which happen with the collusion of one of the internal users, unwittingly or otherwise. These are the attacks with greater chances of success. Attacks such as these only show up after careful analysis of the firewall logs. Again, what you should look for is any unusual activity on your external network as revealed by your reports.

What are the things to consider in relation to attacks?

- Who the persons in charge are and what their responsibilities will be, in the event of an attack. Do you have sufficient on-site skills, or will you refer to a more experienced group in such an event?
- What the immediate course of action will be. Some possibilities are: to block traffic from the offending IP address or subnet, or, in drastic situations, to block all traffic from the firewall completely (this is called *panic* mode, and may be called for if you believe your information is highly sensitive.)
- What the intermediate course of action will be. Some possibilities are: to contact the administrator of the site from which the attack originated, and to maintain a standby status for the duration immediately following the attack.

- What to do in case of a security breach. Even if crackers do find a way through the firewall or around it, it will still have performed its job if you are able to detect the security breach. The only way you will know is by careful analysis of your logs, using it together with other tools or policies that may not even be directly related to the firewall itself. In this case, damage control is your best course of action: seal the hole through which the attackers came through, determine what resources have been stolen or damaged, and assess the extent of the attack.

Shameless plug: the IBM Emergency Response Service is a group which can help you with security-related issues. Their services can be especially helpful in cases of attack recovery and damage control. The IBM ERS Web site is at <http://www.ers.ibm.com>.

3.12 IBM eNetwork Firewall for Windows NT Initial Configuration Parameters

The checklist below is a useful guide. Fill in all the details prior to configuring the firewall. At a single glance, you will find all the information you need when setting up the firewall for the very first time.

Firewall Name	_____
Secure Network Interface Name	_____
Secure Network Interface IP Address	_____
Secure Network Interface Subnet Mask	_____
Non-Secure Network Interface Name	_____
Non-Secure Network Interface IP Address	_____
Non-Secure Network Interface Subnet Mask	_____
Internet Router/Gateway IP Address	_____
Secure Domain Name	_____
Public Domain Name	_____
Secure Domain Name Server IP Address	_____
Non-Secure Domain Name Server IP Address	_____
Internal Mail Server Host Name	_____
Internal Mail Server IP Address	_____

3.13 From Here On...

After defining your security policies and planning your installation, you are now ready to proceed with your firewall implementation.

- To see how to install the firewall, see Chapter 4, "Installing the IBM eNetwork Firewall for Windows NT" on page 43.
- To see a case study of a firewall installation and configuration, see Chapter 5, "A Quick and Basic Configuration: A Case Study" on page 77. This case study ties closely with the examples in this chapter.
- To read more about IP filtering rules to set up your connections, see Chapter 8, "IBM Firewall Rule Base" on page 143 and Chapter 9, "Examples of Rules for Specific Services" on page 161.

- To read more about proxy services for planning purposes, see Chapter 13, “Configuring Proxy Services and SOCKS” on page 293.
- To read more about virtual private networks see Chapter 15, “Virtual Private Network” on page 357.

Chapter 4. Installing the IBM eNetwork Firewall for Windows NT

This chapter takes you through the complete installation of the IBM eNetwork Firewall for Windows NT. We start off with the installation and configuration of the base operating system, Windows NT Server 4.0. We then proceed to the installation of the firewall code itself. Finally, we go through the most common initial configuration steps for the firewall.

4.1 The Base Operating System

The base operating system for the IBM eNetwork Firewall for Windows NT is Windows NT Server 4.0. Windows NT Server 4.0 is a multitasking general-purpose operating system developed by Microsoft as the foundation for its enterprise system offerings. Closely associated with the Intel-based PC platform, it supports hundreds of peripheral devices from different hardware vendors.

Note: The IBM eNetwork Firewall for Windows NT will not run on a Windows NT workstation.

The roots of Windows NT Server are in file and print services, positioned as it was initially against Novell NetWare. Over the years, however, it has evolved into a platform for other types of networked applications. One of the strong suits of Windows NT is its support for a wide variety of network protocols. In today's Internet-focused world, the most notable of these would be TCP/IP.

In the context of the IBM Firewall, our primary interest in Windows NT Server 4.0 is its ability to function as an Internet server and TCP/IP router. But because Windows NT Server 4.0 also carries a number of functions that are superfluous and even detrimental to the operation of the firewall, the configuration of the base operating system differs somewhat from the standard installation.

4.2 Installing Windows NT Server 4.0

The installation process for the base operating system for the IBM Firewall for NT pretty much follows the typical installation for Windows NT Server 4.0. In the following sections, we will walk you through the installation and setup of our base operating system and point out some special steps and why we take them.

Before you begin the installation, make sure that all the hardware components are correctly installed on your firewall system. Take special care that at least two network interface cards are installed on the firewall machine. In general, network cards manufactured from 1996 onwards should work. Note also that your screen must support 1024 x 768 resolution or better for the Configuration Client to work properly.

Note: You can also use the Configuration Client with a smaller video resolution, but you may have to adjust the size of the window to see the complete panels.

The entire installation process of Windows NT is aptly called Setup. Microsoft divides Setup into four phases:

Phase 0: Pre-installation, in which you format or prepare the system partition and copy the necessary setup files onto the hard disk.

Phase 1: Gathering Information about your Computer, in which you enter the server's NetBIOS name, licensing mode, and administration passwords.

Phase 2: Installing Windows NT Networking, in which you specify the network adapters and networking services that you want to enable on the server. The IBM eNetwork Firewall for Windows NT 3.3 does not support dial up connections.

Phase 3: Finishing Setup, in which you complete the setup by specifying the time zone, monitor display settings, and emergency repair settings.

The installation process described below assumes that you are starting with the complete media set for Windows NT Server 4.0 installation, that is, the three setup disks and the CD-ROM media. Refer to the installation guide for Windows NT Server 4.0 from Microsoft for more detailed information on the installation process. If you are quite experienced with Windows NT, you may skip to 4.3, "Base Operating System Installation Summary" on page 47.

4.2.1 Phase 0: Pre-installation

Pre-installation starts with the three setup disks which come with the Windows NT Server 4.0 media. Setup Disk 1 loads a minimal NT kernel onto the system, Setup Disk 2 loads the NT registry and device drivers, and Setup Disk 3 contains drivers for SCSI and IDE devices.

The procedure is as follows:

1. Boot the server hardware from Setup Disk 1 to begin the installation process. During pre-installation, Setup will detect the basic hardware devices that you have attached to your system, the most important of which are the mass storage devices.
2. Setup will ask you where you want to install the system partition of Windows NT and present you with options as to what type of file system you want it to be. Your options will be FAT, NTFS, or to leave the selected partition intact. IBM Firewall requires NTFS, so select that as the file system type for your system partition.

NTFS

NTFS, or NT File System, is the file system designed especially for Windows NT. NTFS performs well for large partitions and incorporates several security features not present in the FAT file system for DOS.

3. Following the setup of the partition, you will then be asked to specify the directory of the Windows NT system files. Accept the default \WINNT.

Setup will proceed by performing a physical check of your hard disk and copies onto it the rest of the setup files from the CD-ROM. The steps are trivial and you may simply choose the default settings. Phase 0 ends with a reboot.

4.2.2 Phase 1: Gathering Information about your Computer

After completing the reboot from Phase 0, Setup continues with the familiar Windows GUI.

1. Setup will ask you to fill in your name and organization and the licensing mode for your server. Fill in the pertinent information. Your licensing policy will depend upon the conditions upon which you purchased your copy of Windows NT Server. A *Per Server* licensing would be appropriate; as you are not using Windows NT for its file or print services, an entry of 0 licenses should be acceptable.
2. Setup will then ask you for your computer's NetBIOS name, which can be up to 15 characters long. Fill in this entry with the NetBIOS name you want associated with the firewall. This is especially important if you will later make the firewall a member of a domain. See Chapter 13, "Configuring Proxy Services and SOCKS" on page 293 for more information.
3. Choose **Stand-Alone Server** when Setup asks you what type of server you are installing.

PDC and BDC

The two other options, Primary Domain Controller and Backup Domain Controller, are used for maintaining and managing user accounts from a central database; IBM eNetwork Firewall for Windows NT performs neither function.

4. Enter an initial Administrator password when Setup prompts you. The password is case sensitive and can be up to 14 characters long.
5. Setup will prompt you for whether you wish to create an Emergency Repair Disk. The Emergency Repair Disk will copy important registry and configuration files to a backup floppy. Creating the Emergency Repair Disk is not absolutely necessary at this time.

Emergency Repair Files.

The files to be written to the Emergency Repair Disk are also found in the `\WINNT\REPAIR` directory of the system partition.

To create an Emergency Repair Disk, run the command `rdisk.exe` from the Windows NT command prompt or the Run... dialog box.

6. Setup will ask you if you want to specify optional components. These components refer to Windows NT accessories, bitmaps, and screen savers. None of these will be essential, and it would be a good idea to deselect all the options listed.

4.2.3 Phase 2: Installing Windows NT Networking

Setup will now install the networking components of Windows NT. For security purposes, it is important that the firewall server *not* be connected to the non-secure network at this time.

1. Setup will ask if the server will participate in a network and if so, whether it will be wired to the network or will connect to it through a modem. The configuration intended for the IBM Firewall will be a direct network connection for all interfaces.

2. Setup will ask you if you want to install the Microsoft Internet Information Server on the system. This service is unnecessary and presents some security risks for the firewall; therefore, it should not be loaded at all.
3. The next screen will ask if you want Setup to automatically detect your network adapter cards. Click **Yes**.
4. As the firewall will require at least two adapters, you should ask Setup to also detect additional network adapter cards by clicking on **Find Next**.
5. If your network adapter cards cannot be detected, you can manually specify them from the Select from List dialog or from the Have Disk dialog box if special drivers come with the adapters you are using.
6. Setup will now ask you for the network protocols that you will be using. Select only **TCP/IP** from the list and disable all others.
7. Setup will show you a screen of the default network services that it will install. These network services are: NetBIOS Interface, RPC Configuration, Server, and Workstation. Setup will not give you any choice as to removing these services, so simply proceed with the process by clicking on **Next**.
8. Setup will proceed to configure the network adapters you specified earlier. Depending on the type of adapter card you are using, you will be prompted for additional information such as interrupts and memory locations.
9. TCP/IP configuration comes next. Setup will ask if you wish to configure the interfaces using DHCP. Click on **No** because the IP addresses on the firewall's interfaces must be fixed and well-known. Instead, proceed to fill in the IP address, subnet mask, and gateway information of each network adapter. You may leave the other information such as DNS, WINS, and Routing blank at this time.

In leaving the TCP/IP configuration screen, Setup will warn you that your WINS is not configured. You may safely ignore this message.
10. The next screen will show you the service, protocol, and adapter bindings. Bindings show the hierarchical relationship from the physical network adapter to the drivers to the software interfaces of Windows NT. Accept the defaults at this time.
11. Setup will ask you for the domain or workgroup that the firewall server will join. Select **Workgroup** and type in a name. In reality, the firewall will not need this information because it will not participate in a Microsoft NetBIOS network (unless you make it part of a Microsoft domain -- see 13.1.6, "Configuring for NT Logon Password Authentication" on page 302), but Windows NT will insist that these fields be filled in. An uncommon hard-to-guess combination of letters and numbers would be ideal.

4.2.4 Phase 3: Finishing Setup

Phase 3 of the installation is almost trivial in comparison with the previous steps taken in the setup process.

1. Setup will ask you to set the server's current date, time, and time zone. Enter the correct information here as some time-sensitive applications operating either inside or outside the secure network may not work with the firewall if the date and time ranges do not match.
2. Setup will attempt to detect the display adapter for your system. The firewall Configuration Client requires a minimum resolution of 1024 x 768 to work

properly. Set the resolution to this value or higher. Part of this installation process will include a test of the resolution and display on your computer screen.

3. If you asked Setup to create an Emergency Repair Disk in Phase 1 of the installation, you will be prompted to insert a blank floppy into the disk drive. Setup will copy the recovery information into this floppy.
4. Having completed all the phases of Setup, you will now be asked to reboot the system.

4.3 Base Operating System Installation Summary

Below is a summary of the significant steps in the installation of Windows NT on the firewall system. Use it as a checklist to confirm the options and settings of your base operating system installation.

- Disconnect the firewall system from the network during this installation.
- Format the system partition of the firewall system as an NTFS partition.
- If you are unsure as to what licensing policy to use, select **Per Server** with 0 client licenses.
- Install Windows NT Server as a Stand-Alone Server on the firewall system.
- You do not need to create an Emergency Repair Disk at this time.
- Do not install any of the optional Windows NT components on the firewall system.
- Do not install the Microsoft Internet Information Server.
- Make sure that all the network cards you will be using for the firewall are detected during setup and that all of these are configured properly.
- Select only **TCP/IP** as the network protocol to use on the firewall system. Disable or deselect all other network protocols.
- Specify the IP address, subnet mask, and gateway of each network interface of the firewall system. Leave the other information such as DNS, WINS, or DHCP empty.
- Select **Workgroup** as the network type that the firewall system will participate in. A fictitious name will do as the firewall system will not actually be joining a Microsoft network.

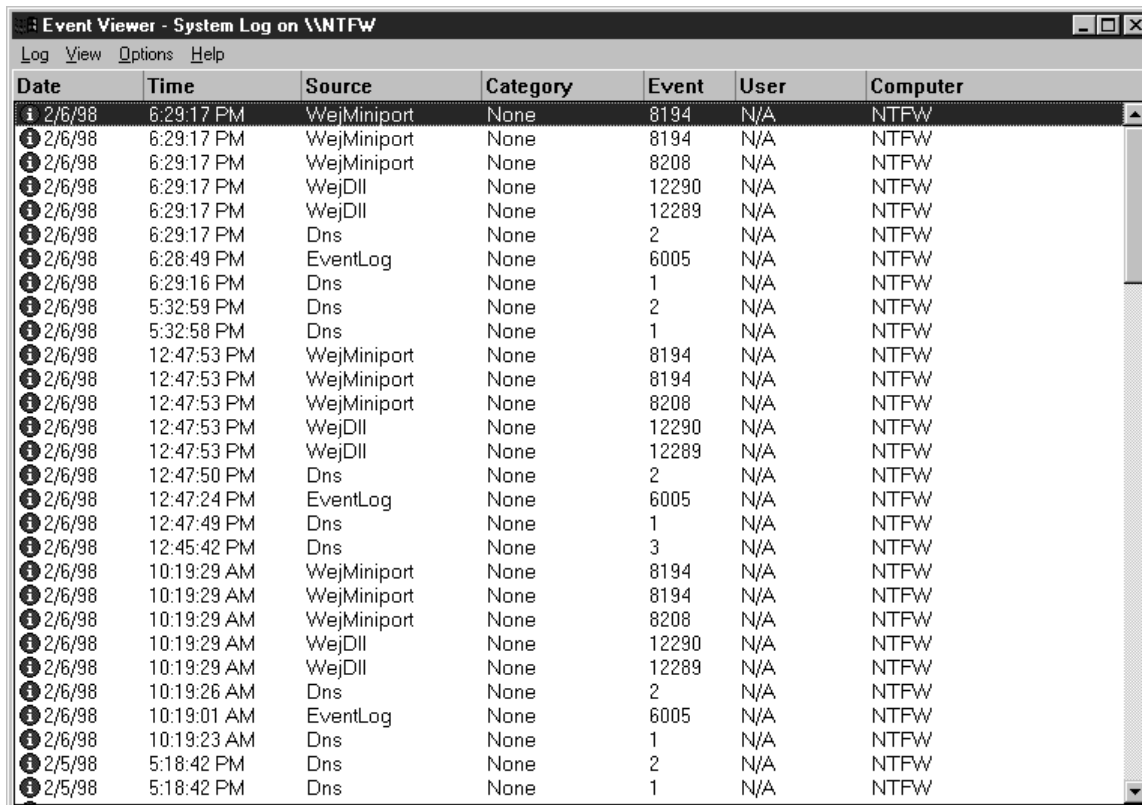
4.4 Post-Installation Procedures

To ensure a smooth installation of the IBM eNetwork Firewall for Windows NT, you must make sure that your base system boots correctly, with all of the specified software services enabled and all hardware devices recognized. You must also install the Microsoft DNS Server component and the latest NT service pack and hot fixes to correct some security deficiencies in Windows NT. Finally, you will also want to manually check the security of your installation prior to the hardening that the firewall code installation actually performs.

4.4.1 Checking Your Installation

Log in to the base firewall system as Administrator. Make sure that all of the Windows NT services have been started. You can do this by running the Event Viewer.

1. Click on **Start** on the NT task bar.
2. Select **Programs**→**Administrative Tools (Common)** and click on **Event Viewer**.
3. Watch out for any critical stops logged in the Event Viewer window. These will be highlighted with a red Stop sign. You must resolve these errors before proceeding with the installation of the firewall.



The screenshot shows the Event Viewer window titled "Event Viewer - System Log on \\NTFW". The window contains a table with the following columns: Date, Time, Source, Category, Event, User, and Computer. The table lists various system events, including network-related events (WejMiniport, WejDll) and system events (Dns, EventLog). The events are sorted by time, showing a sequence of operations from 5:18:42 PM on 2/5/98 to 12:47:53 PM on 2/6/98.

Date	Time	Source	Category	Event	User	Computer
2/6/98	6:29:17 PM	WejMiniport	None	8194	N/A	NTFW
2/6/98	6:29:17 PM	WejMiniport	None	8194	N/A	NTFW
2/6/98	6:29:17 PM	WejMiniport	None	8208	N/A	NTFW
2/6/98	6:29:17 PM	WejDll	None	12290	N/A	NTFW
2/6/98	6:29:17 PM	WejDll	None	12289	N/A	NTFW
2/6/98	6:29:17 PM	Dns	None	2	N/A	NTFW
2/6/98	6:28:49 PM	EventLog	None	6005	N/A	NTFW
2/6/98	6:29:16 PM	Dns	None	1	N/A	NTFW
2/6/98	5:32:59 PM	Dns	None	2	N/A	NTFW
2/6/98	5:32:58 PM	Dns	None	1	N/A	NTFW
2/6/98	12:47:53 PM	WejMiniport	None	8194	N/A	NTFW
2/6/98	12:47:53 PM	WejMiniport	None	8194	N/A	NTFW
2/6/98	12:47:53 PM	WejMiniport	None	8208	N/A	NTFW
2/6/98	12:47:53 PM	WejDll	None	12290	N/A	NTFW
2/6/98	12:47:53 PM	WejDll	None	12289	N/A	NTFW
2/6/98	12:47:50 PM	Dns	None	2	N/A	NTFW
2/6/98	12:47:24 PM	EventLog	None	6005	N/A	NTFW
2/6/98	12:47:49 PM	Dns	None	1	N/A	NTFW
2/6/98	12:45:42 PM	Dns	None	3	N/A	NTFW
2/6/98	10:19:29 AM	WejMiniport	None	8194	N/A	NTFW
2/6/98	10:19:29 AM	WejMiniport	None	8194	N/A	NTFW
2/6/98	10:19:29 AM	WejMiniport	None	8208	N/A	NTFW
2/6/98	10:19:29 AM	WejDll	None	12290	N/A	NTFW
2/6/98	10:19:29 AM	WejDll	None	12289	N/A	NTFW
2/6/98	10:19:26 AM	Dns	None	2	N/A	NTFW
2/6/98	10:19:01 AM	EventLog	None	6005	N/A	NTFW
2/6/98	10:19:23 AM	Dns	None	1	N/A	NTFW
2/5/98	5:18:42 PM	Dns	None	2	N/A	NTFW
2/5/98	5:18:42 PM	Dns	None	1	N/A	NTFW

Figure 17. The Event Viewer

We recommend that you perform this checking procedure after every hardware or software installation on the firewall system so you can identify problem areas at an early stage.

4.4.2 Setting System Resolution

The Configuration Client of IBM eNetwork Firewall for Windows NT requires a minimum resolution of 1024 x 768. If you have not already done so, open **Display Properties of Control Panel** and select the most appropriate screen size available.

4.4.3 Installing the Microsoft DNS Server

The IBM Firewall uses the Microsoft DNS Server to provide Internet naming services. You install the Microsoft DNS Server through the Services panel of the Network configuration windows of Control Panel.

1. Click on the **Start** button on the task bar. Go to **Settings** and click on **Control Panel**.



Figure 18. The Control Panel

2. Double-click on the **Network** object of the Control Panel.
3. Click on the **Service** tab of the Network dialog. You should see a list of the existing Network Services.

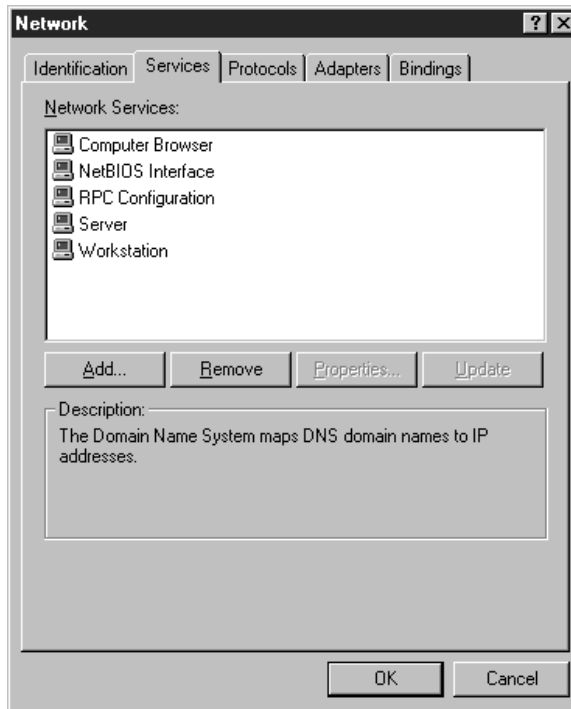


Figure 19. The Network Services Panel

4. Click on the **Add...** button to bring up a list of additional network services.
5. Choose **Microsoft DNS Server** from the list and click on OK to confirm the choice. The NT installation media must be in the CD-ROM drive at this time.

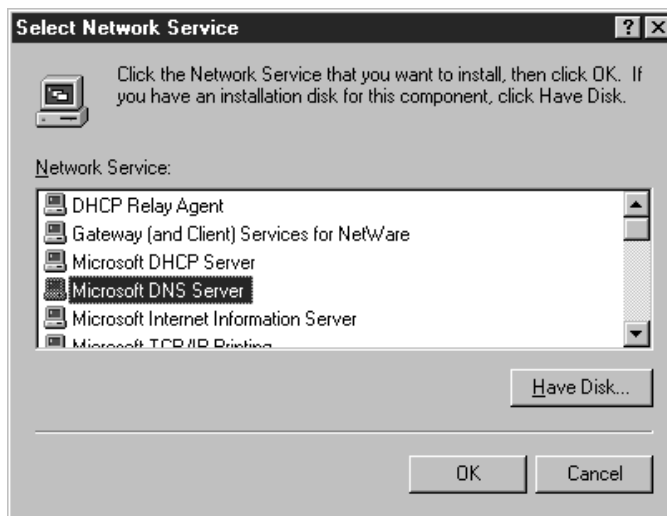


Figure 20. Installing Microsoft DNS Server

Windows NT will ask you to reboot your system after the Microsoft DNS Server installation process completes. After rebooting, check the Event Viewer and the Services list in Control Panel to ensure that all devices and processes have started correctly.

Installing the Microsoft DNS Server will add DNS Manager to your Administrative Tools (Common) program group. DNS Manager is a graphical utility for configuring

DNS servers on a Windows NT Server system. You can access the DNS Manager by selecting **Start**→**Administrative Tools (Common)**→ **DNS Manager**.

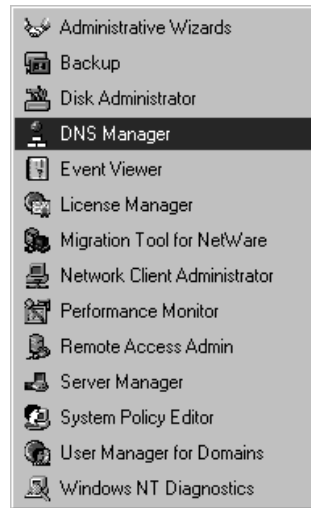


Figure 21. Running DNS Manager from Administrative Tools (Common)

4.4.4 Installing Service Pack 3 and Post-SP3 Hot Fixes

The IBM eNetwork Firewall for Windows NT requires SP3 plus the hot fixes mentioned below. You may also attain this service level by installing SP4 or SP5 (the latest service pack available at this writing).

Note: An IBM eNetwork Firewall for Windows NT 3.3 installed with SP4 or SP5 is also supported by IBM.

SP3 updates certain files and enhances security on both the Windows NT Workstation 4.0 and the Windows NT Server 4.0. The Post-SP3 Hot Fixes directly address security issues relating to the TCP/IP implementation on NT, pending the next service pack from Microsoft. You can download the current service pack and the hot fixes from the Microsoft FTP site.

Below is a brief description of Service Pack 3 and the currently required hot fixes. They must be installed in the order given below.

- `nt4sp3_i.exe`. Download from <ftp://ftp.microsoft.com/bussys/winnt/winnt-public/fixes/usa/nt40/ussp3/i386/>. Service Pack 3 updates the RPC transport and installs several drivers for DirectX and ODBC. It patches support for SMB signing, password filtering, and restrictions for anonymous user access. For more information, please see the README.TXT included with the service pack.

Service packs change only those files that were originally set up on the Windows NT system. Service pack releases are cumulative, containing all the fixes included in previous service packs.

- `ndisfixi.exe`. Download from <ftp://ftp.microsoft.com/bussys/winnt/winnt-public/fixes/usa/nt40/hotfixes-postSP3/ndis-fix/>. `ndis-fix` patches memory leaks and the dreaded "blue screen of death" in Windows NT arising from the use of intermediate NDIS miniport drivers. For more information, please see the README.TXT included with the hot fix.

- `dnsfixi.exe`. Download from <ftp://ftp.microsoft.com/bussys/winnt/winnt-public/fixes/usa/nt40/hotfixes-postSP3/dns-fix/>. `dns-fix` insulates the Microsoft DNS Server from a malicious denial-of-service attack which causes an Access Violation in the DNS service. For more information, please see the README.TXT included with the hot fix.
- `chargeni.exe`. Download from <ftp://ftp.microsoft.com/bussys/winnt/winnt-public/fixes/usa/nt40/hotfixes-postSP3/simptcp-fix/>. `simptcp-fix` modifies the NT TCP/IP to be more attack resistant. In particular, it resolves the problem of an increase in bandwidth utilization when a computer comes under attack. For more information, please see the README.TXT included with the hot fix.
- `tearfixi.exe`. Download from <ftp://ftp.microsoft.com/bussys/winnt/winnt-public/fixes/usa/nt40/hotfixes-postSP3/teardrop2-fix/>. `teardrop2-fix` patches NT to protect itself from a modified version of the *teardrop attack*. In this type of attack, NT will hang after receiving a number of deliberately corrupted UDP packets. SP3 already modifies NT to reject the original teardrop attack; this fix is provided to meet a variant. For more information, please see the README.TXT included with the hot fix.

To install the service pack and the hot fixes, simply run their associated .EXE files. You can do this from either the Command Prompt or from the Windows Explorer. The patches are self-extracting and self-installing. Be prepared to reboot after every install.

Check the IBM Firewall home page (<http://www.software.ibm.com/enetwork/firewall>) regularly for updates on new required software patches for Windows NT and the IBM eNetwork Firewall for Windows NT. Do not indiscriminately load new hot fixes from Microsoft without first verifying with IBM as the updates may generate incompatibilities with the IBM Firewall code.

4.5 Enhancing Security on the Base Operating System

In installing the IBM eNetwork Firewall for Windows NT code, you will be given the opportunity to perform *hardening* on your base system. Hardening refers to the process of disabling unnecessary services and user accounts on the operating system. However, additional steps that you take to secure the base operating steps will greatly fortify the firewall system before the hardening process.

4.5.1 Physical Security

Secure the firewall system by placing it in a room where access is strictly controlled. NT is most vulnerable if a person of malicious intent has physical access to the system.

4.5.2 Account Security

The Administrator account is well-known for Windows NT systems and is therefore a favorite target for password-guessing attacks by crackers. Change (do not delete!) the name of the Administrator account from the User Manager and create a decoy account named Administrator which has no privileges at all.

In this example, we change the name of Administrator to Root. We also create a decoy Administrator account that has only guest privileges.

1. Select **Start**→**Programs**→**Administrative Tools (Common)**→**User Manager for Domains**.
2. Highlight the Administrator account.
3. Click on **User** from the pull-down menu and click on **Rename....** This brings up the Rename dialog box.

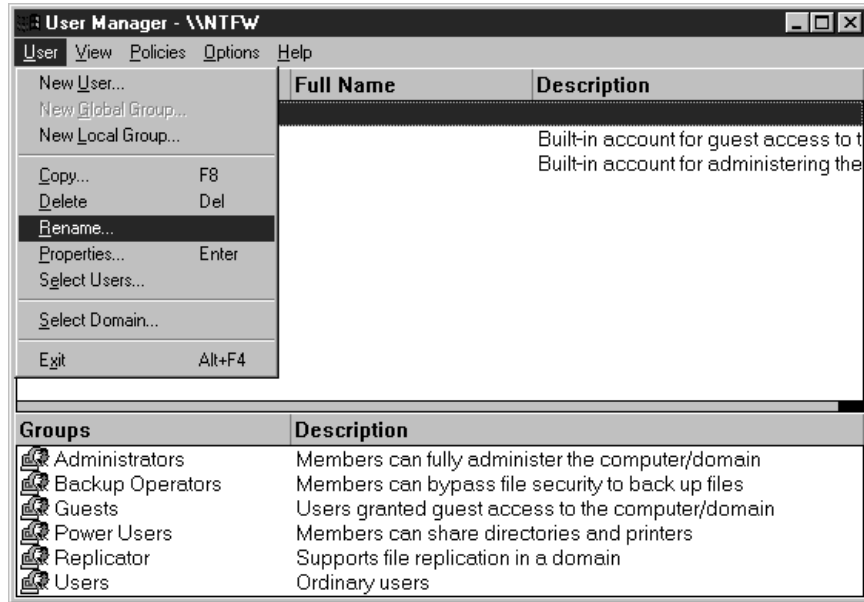


Figure 22. Renaming an Account

4. In the Change To field, enter the new user name. In this redbook's examples, we use Root as the substitute name for Administrator. Click on **OK** to confirm.

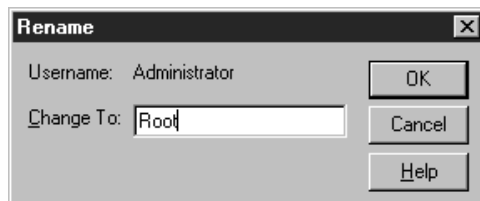


Figure 23. The Rename Dialog from User Manager

To create a dummy account:

1. From the User pull-down menu, click on **New User**.
2. Create another account named Administrator. Enter other pertinent information such as the password and the full name.
3. Remove the decoy Administrator's membership in the Users group and grant it membership in Guests instead. To do this, click on **Groups** to bring up the Group Memberships dialog. You can manipulate memberships by clicking on **<-Add and Remove->..**

Windows NT remembers the last account you logged on as, and prominently displays this in the logon screen. You do not want it to do this, otherwise your efforts at creating the dummy account will come to naught. To disable this feature:

1. Click on **Start**→**Administrative Tools (Common)**→**System Policy Editor**.

2. From the System Policy Editor's menu bar, open **File** and click on **Open Registry**. This will bring up the two icons in the System Policy Editor's space as shown in Figure 24 on page 54.

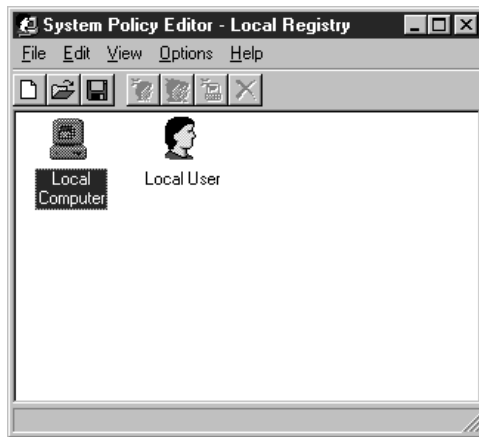


Figure 24. The System Policy Editor

3. Double-click on the **Local Computer** object. This will bring up the dialog in Figure 25.

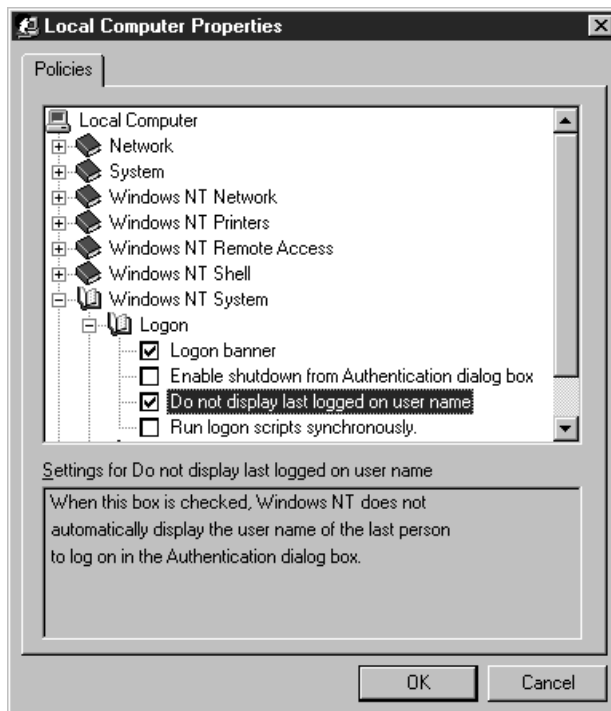


Figure 25. Local Computer Logon Policies

4. Open the object **Windows NT System** and the **Logon** object in it.
5. Check the **Do not display last logged on user name** check box. Click on **OK** to confirm the change and go back to the System Policy Editor.
6. On the main window of the System Policy Editor, open the **File** pull-down menu again and click on **Save**.
7. Close the System Policy Editor.

4.5.3 User Rights Policy

In the User Manager for Domains dialog, click on the **Policies** pull-down menu and select **User Rights....** Check the **Show Advanced User Rights** box. Look at each user right and verify to which group it is assigned to. In principle, only the Administrators group should have any rights at all on the firewall, so you might want to remove the other groups from all the rights lists.

It is imperative that you remove the group Everyone from all the rights lists. For example, one of the things you should look out for will be the right 'Access this computer from the network'. A quirk of Windows NT is that it gives the user group Everyone this privilege by default.

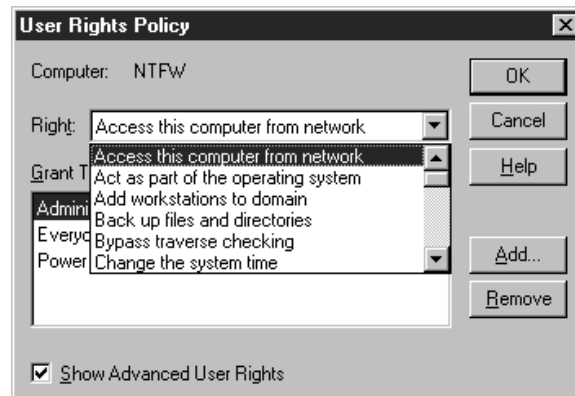


Figure 26. The User Rights Policy Dialog from User Manager

4.5.4 Disable Shares

By default, Windows NT Server shares the root directory of each partition and the directory containing Windows NT system files. To remove these shares, follow the steps below:

1. From the NT desktop, double-click on the **My Computer** icon to open it.
2. Highlight drive C: of in the My Computer object group. Click on **File** from the pull-down menu and select **Sharing....** This will open the Properties dialog for drive C:.

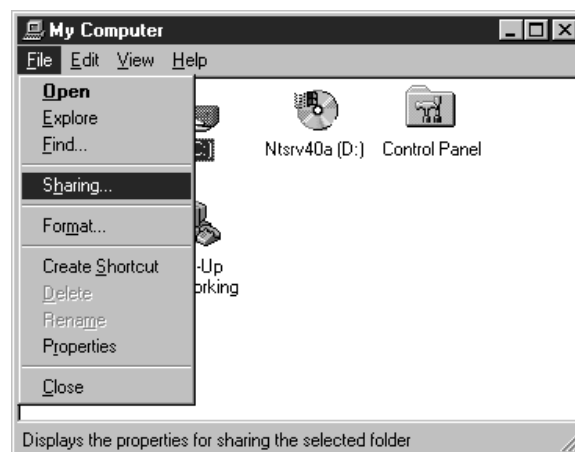


Figure 27. The Sharing... Option for Drives

3. Click on the **Not Shared** radio button to remove the share. Click on **OK** to close the dialog.

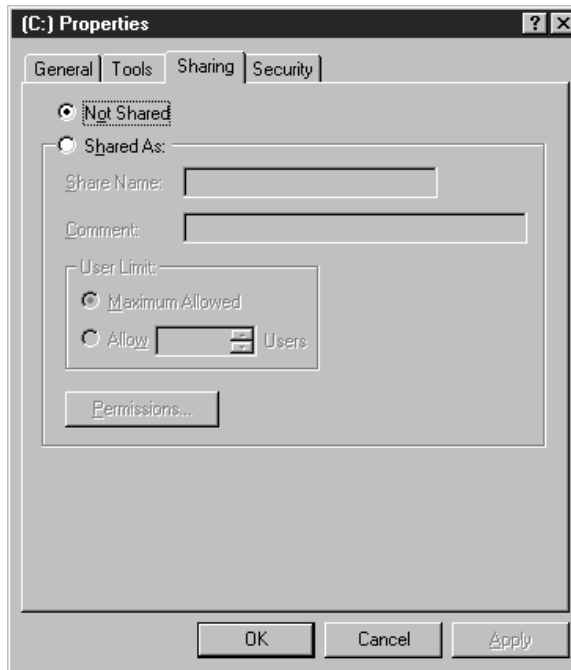


Figure 28. Properties for Drive C:

4.5.5 Connectivity Test

As a last step prior to installing the base firewall code, test the connectivity of the firewall machine to the network. Connect both interfaces to their respective networks, and perform ping reachability tests to the firewall from both sides.

Disconnect the firewall from the network once you've established that both interfaces are active.

4.6 Installing the Base Firewall

At this point, you must have finished installing the base operating system for the firewall. It is of utmost importance that the network interfaces of the firewall system be properly configured and the service pack and hot fixes be installed on the Windows NT Server.

To install the IBM eNetwork Firewall for Windows NT, you must enable your firewall system to act as a router. This is a native function of the multi-homed Windows NT Server. The routing functions are enhanced by the IBM Intermediate Support Driver, a packet filtering protocol that works with the firewall code.

The IBM eNetwork Firewall for Windows NT setup program will install the firewall server code and the Firewall Configuration Client components onto the system. The actual firewall server code consists of special software services. The Firewall Configuration Client gives you a GUI interface from which to manage the firewall, SOCKS monitoring agent for monitoring active SOCKS connections and Netscape Navigator 4.0 for reading the help files that come with the package.

4.6.1 Installation Media

IBM eNetwork Firewall for Windows NT is available on CD-ROM installation media. Insert this into your CD-ROM drive.

For the purposes of this redbook, drive E: designates the location of the IBM eNetwork Firewall for Windows NT CD-ROM install media.

4.6.2 Logging On

The user account you use to install IBM eNetwork Firewall for Windows NT must be a member of the Administrators group of your Windows NT system.

4.6.3 Activating Windows NT's IP Forwarding Function

The firewall functions to some extent as an Internet router. To enable this capability, you must activate Windows NT's IP forwarding function. Follow the steps outlined below:

1. Click on the **Start** button of the taskbar and go to **Settings**→**Control Panel**. Double-click on the **Network** object to open the Network dialog box.
2. Go to the **Protocols** tab and highlight TCP/IP Protocol. Click on **Properties** to bring up the Microsoft TCP/IP Properties dialog box.

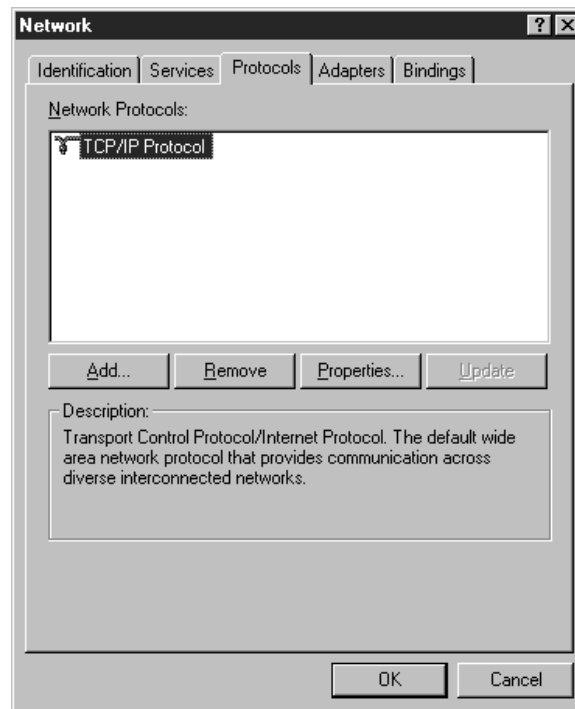


Figure 29. The Protocols Tab in the Network Dialog

3. The Microsoft TCP/IP Properties dialog will have five tabs: IP address, DNS, WINS Address, DHCP Relay, and Routing. Click on the **Routing** tab, and check the **Enable IP Forwarding** box. Click on **OK** to commit the changes and close the Microsoft TCP/IP Properties dialog.

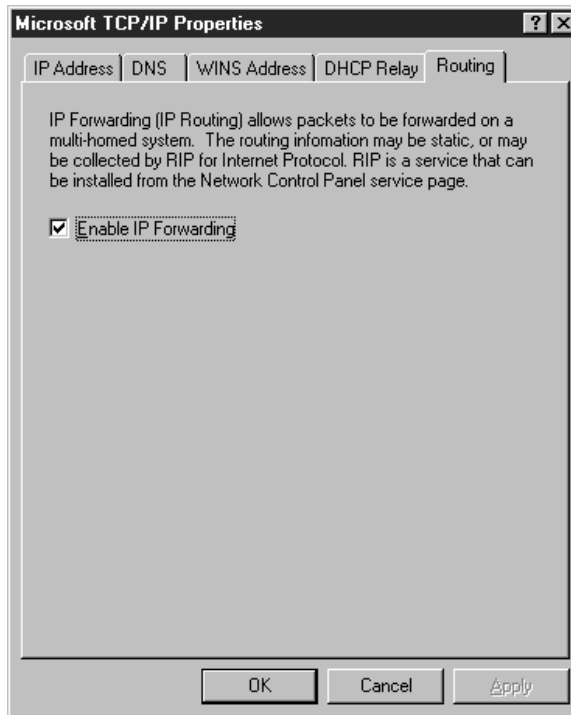


Figure 30. Enabling Routing for Windows NT Server

4. You will end up back at the Network dialog box. Do not close the dialog, but continue to the next section.

Note: If you know your firewall never uses routed packets (that is, all the filter rules have routing = local), then you can disable IP Forwarding for additional security.

4.6.4 Installing the IBM Intermediate Support Device Driver

The IBM Intermediate Support Driver sits in the logical link layer between Windows NT's TCP/IP code and the network adapter device drivers. It enhances the security on the network layer by performing packet filtering functions. It also binds to the logging facility of Windows NT so you can see the events related to your network interface adapters. The code for this is included in the IBM Firewall distribution media. To install the driver, follow the steps outlined below:

1. Go to the **Protocols** tab of the Network dialog box.
2. Click on the **Add...** button to bring up the Select Network Protocol dialog box.

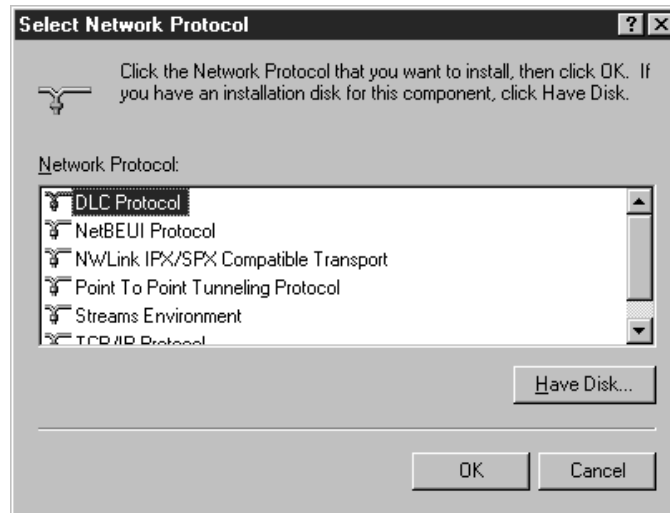


Figure 31. Selecting a Network Protocol to Add to Windows NT

3. Click on **Have Disk...** and specify the location of your IBM Firewall for NT setup files. This brings up the Select OEM Option dialog box.

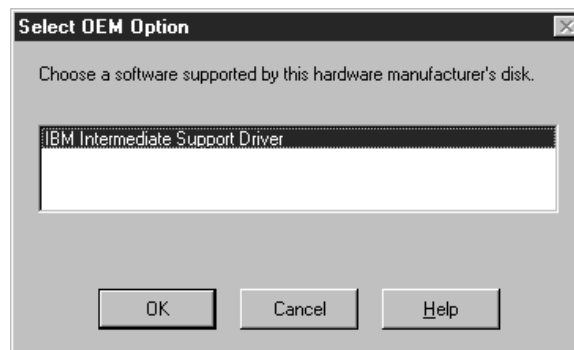


Figure 32. Adding the IBM Intermediate Support Driver

4. On the Select OEM Option dialog, select **IBM Intermediate Support Device Driver** and click **OK**. Close the Network dialog. Windows NT will prompt you to reboot, but do not restart your computer at this time as you still need to install the IBM eNetwork Firewall for Windows NT.

4.6.5 Running the Installation Program

The installation program will install the firewall code, the firewall Configuration Client, the SOCKS monitor, and Netscape Navigator 4.0. It is pretty straightforward from here on, as the install process is initiated from a single InstallShield program.

The InstallShield program is called setup.exe and is found in the appropriate setup directory of the international language you are installing for. In our examples, we simply assume that it is found in the root directory of our installation CD.

1. Click on the **Start** icon of the task bar and select **Run....** Type in `e:\en_us\setup.exe` and click on **OK**.



Figure 33. The IBM e-Net Network Firewall for Windows NT Installation's Welcome Screen

2. After the introductory screen, you will be given the option to install Netscape Navigator 4.0. This is not the full Netscape Communicator suite, just the Web browser component. You will need this to read the help files of IBM Firewall for NT. Simply accept the default settings. The IBM Firewall for NT setup program continues after installing Navigator.

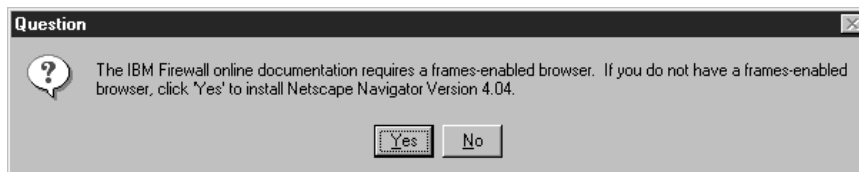


Figure 34. Installing Netscape Navigator 4.0

3. The IBM Firewall Installation Options dialog comes up next. Select from the list which components you would like to install. The IBM Firewall Program files are required unless you have a previous installation. For a basic installation, you should also install the firewall reference books. Move to the next screen by clicking on **Next**.



Figure 35. The IBM eNetwork Firewall for Windows NT Installation Options

4. Double-check your configuration and click **Next** if you are satisfied with it. Setup will start copying files.
5. The install program will ask you if you want firewall hardening performed on your system. The hardening process will disable unnecessary services on your firewall system. The procedure will take some time for the install program to complete, but it is a worthwhile exercise nevertheless. Click on **Yes** to proceed.

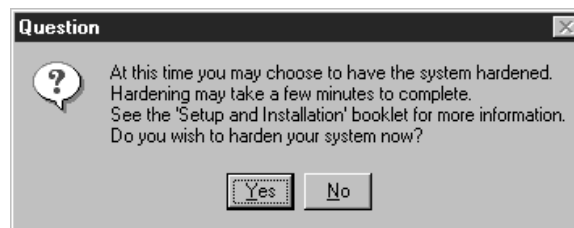


Figure 36. IBM eNetwork Firewall for Windows NT Hardening Prompt

Section 4.7, “Changes Made by the Hardening Process” on page 62 explains the results in greater detail.

Once the installation completes, you will have two additional program groups in your Start menu. These are IBM Firewall and Netscape Navigator.

The IBM Firewall program group gives you five items: Configuration Client, Reference, SOCKS Monitor, Uninstall, and User's Guide.



Figure 37. The IBM eNetwork Firewall for Windows NT Program Group

Having completed all these steps, reboot the firewall machine to enable all the changes.

4.6.6 Connect the Firewall to the Network

Having completed the installation of the firewall code, the firewall machine can now be considered secure. The IBM Firewall starts out its life denying all connections to both its interfaces. In the configuration process outlined in Chapter 8, "IBM Firewall Rule Base" on page 143, Chapter 13, "Configuring Proxy Services and SOCKS" on page 293, and the sections that follow, you will see how to activate these connections.

On the same note, you will not be able to ping the firewall now because it denies all ICMP packets until you specifically enable it to respond. If your connectivity test in 4.5.5, "Connectivity Test" on page 56 gave you positive results, and if the NT Event Viewer reports no errors, you should have no cause for concern.

4.7 Changes Made by the Hardening Process

The hardening process disables Windows NT services that will not be used by the firewall. Below is a table of services before and after the installation and hardening.

Table 9 (Page 1 of 2). Services Before and After the Installation of IBM eNetwork Firewall for Windows NT

Service	Before Installation	After Installation
Alerter	Automatic	Disabled
ClipBook Server	Manual	Disabled
Computer Browser	Automatic	Disabled
DHCP Client	Disabled	Disabled
Directory Replicator	Manual	Disabled
EventLog	Automatic	Automatic
IBM Firewall Initialization Service	Not present	Automatic
IBM Firewall Configuration Service	Not present	Automatic
IBM Firewall Filter Log Service	Not present	Automatic
IBM Firewall Log Daemon	Not present	Automatic
IBM Firewall Proxy FTP Daemon	Not present	Automatic
IBM Firewall Proxy HTTP Daemon	Not present	Automatic
IBM Firewall Proxy Telnet Daemon	Not present	Automatic
IBM Firewall Secure Mail Proxy Server	Not present	Automatic
IBM Firewall SOCKS Monitor Service	Not present	Manual
IBM Firewall SOCKS Server	Not present	Automatic
License Logging Service	Automatic	Disabled
Messenger	Automatic	Disabled
Microsoft DNS Server	Automatic	Automatic
Net Logon	Manual	Disabled
Network DDE	Manual	Disabled

Table 9 (Page 2 of 2). Services Before and After the Installation of IBM eNetwork Firewall for Windows NT

Service	Before Installation	After Installation
Network DDE DSDM	Manual	Disabled
NT LM Security Support Provider	Manual	Manual
Plug and Play	Automatic	Automatic
Remote Procedure Call (RPC) Locator	Manual	Disabled
Remote Procedure Call (RPC) Service	Automatic	Automatic
Schedule	Manual	Manual
Server	Automatic	Disabled
Spooler	Automatic	Disabled
TCP/IP NetBIOS Helper	Automatic	Disabled
Telephony Service	Manual	Disabled
UPS	Manual	Disabled
Workstation	Automatic	Disabled

If the firewall is being installed on a machine that is a member of an NT domain, hardening does not disable the Net Logon, TCP/IP NetBIOS Helper and Workstation services.

4.8 Initial Configuration of IBM eNetwork Firewall for Windows NT

Now that you've installed the firewall and other prerequisite software, you will want to perform some initial configuration. The preliminary configuration steps performed below will apply to all but the most specialized implementations of IBM Firewall for NT.

A wizard has been provided which enables a user who does not have extensive knowledge of the firewall to have a basic firewall configuration up and running quickly after installation. The usage of the wizard is discussed in Chapter 6, "Firewall Setup Using the Wizard" on page 109.

Your first step will be to verify your TCP/IP configuration. Next, you will assign your network interfaces to the secure and non-secure networks, configure the DNS cache and mail forwarding on the firewall. Finally, you will define some common network objects.

4.8.1 Prerequisites

Refer back to the network planning sheet you completed in Chapter 3, "Planning Your Firewall Installation" on page 29. For the initial configuration, you will need information on the following:

Firewall Name _____

Secure Network Interface Name _____

Secure Network Interface IP Address _____

Secure Network Interface Subnet Mask _____

Non-secure Network Interface Name _____

Non-secure Network Interface IP Address _____

Non-secure Network Subnet Mask _____

Internet/Router Gateway IP Address	_____
Secure Domain Name	_____
Public Domain Name	_____
Secure Domain Name Server IP Address	_____
Non-secure Domain Name Server IP Address	_____
Internal Mail Server Host Name	_____
Internal Mail Server IP Address	_____

The values we use to fill in these details in the subsequent screens will be the same values used for the example in Chapter 5, “A Quick and Basic Configuration: A Case Study” on page 77.

4.8.2 Checking Your TCP/IP Configuration

Verify the TCP/IP configuration on the firewall system. Make sure that you have the correct IP addresses for both the interfaces and the Internet gateway.

1. Click on **Start** and open **Settings**→**Control Panel**. Double-click on the **Network** object to bring up the Network dialog box.
2. Click on the **Protocols** tab and highlight **TCP/IP Protocol** from the resulting list. Click on **Properties...** to bring up the Microsoft TCP/IP Properties dialog.
3. Click on the **IP Address** tab. Verify the IP address of each adapter. Open the **Adapter** drop-down list to view each adapters settings.
4. Verify the default gateway for each adapter. Make sure you have a Default Gateway entry to your Internet router for the designated non-secure adapter.

Alternatively, you can also check the IP settings using the IPCONFIG.EXE from the command prompt.

1. Click on **Start** and open **Programs**→ **Command Prompt**.
2. From the command window, type IPCONFIG.

```

Microsoft(R) Windows NT(TM)
(C) Copyright 1985-1996 Microsoft Corp.

C:\>IPCONFIG

Windows NT IP Configuration

Ethernet adapter Wej_IbmTok42:

    IP Address. . . . . : 10.0.0.1
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . :

Token Ring adapter Wej_IbmTok41:

    IP Address. . . . . : 204.189.54.97
    Subnet Mask . . . . . : 255.255.255.248
    Default Gateway . . . . . : 204.189.54.102

C:\>

```

Figure 38. Checking for the IP Address and Gateway Settings

IPCONFIG returns the IP address, subnet mask, and default gateway for each adapter.

4.8.3 Starting the Configuration Client

The Configuration Client is the control center for IBM Firewall for NT. From here, you can configure almost any aspect of the operation of the firewall. You can also view the firewall logs from the Configuration Client, as well as launch the firewall's user guide and reference guide.

Only members of the Administrators group of the NT system and designated firewall administrators can log on to the Configuration Client.

To start the Configuration Client and log on to it:

1. Click on **Start** and open the **Programs**→**IBM Firewall** program group. Click on the **Configuration Client** object to start the program.
2. You will be presented with a logon dialog. If the Configuration Client is running on the same machine as the server, select **Local** as your Logon Type.

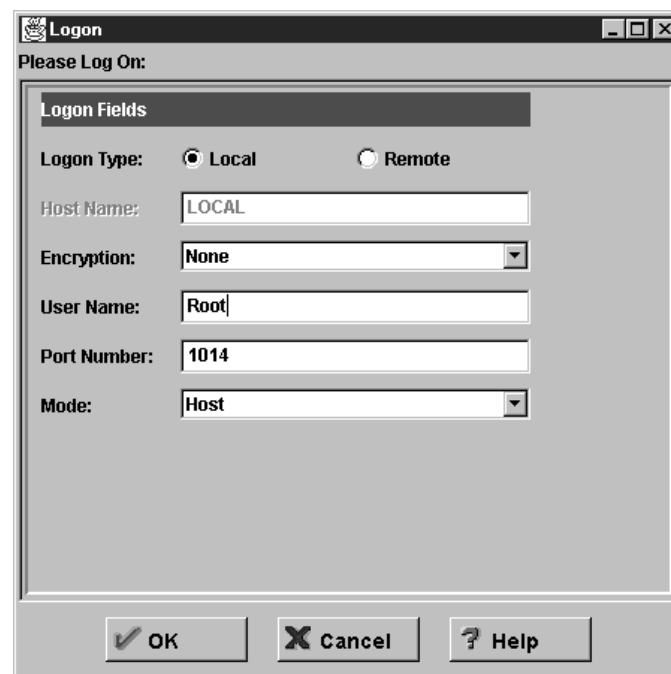


Figure 39. The Firewall Configuration Client's Welcome Challenge

3. For a local logon, the Host Name field will be locked at LOCAL.
4. For a local logon, accept None for the Encryption type.
5. The User Name will be set to the user account name which you used to log on. This account should have Administrator group privileges on the firewall server.
6. Keep the Port Number at 1014. This is the default configuration port setting for the Configuration Client.
7. Select **Host** as the Mode for the logon. Host mode is valid if you are configuring only the firewall that you are logged on to. The other option, Enterprise, will be used only for administering managed firewalls from an Enterprise Firewall machine.
8. Click on **OK** to accept the logon parameters. The Authentication dialog box will appear.

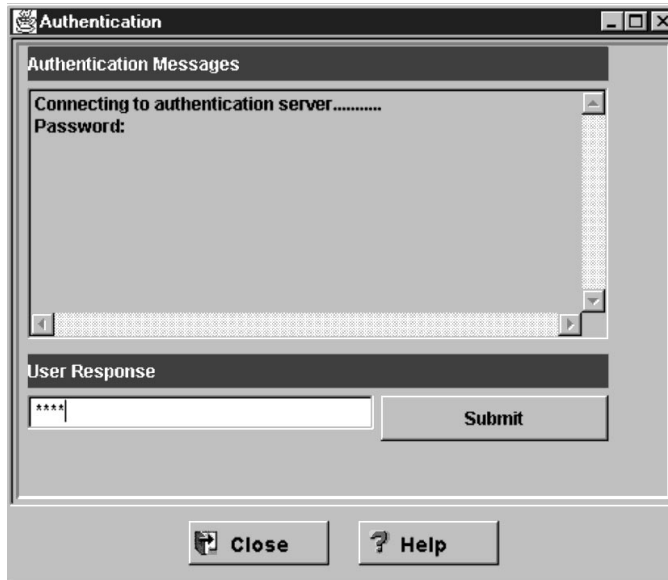


Figure 40. The Configuration Client's Authentication Dialog Box

9. Enter the administrator account name's password in the User Response field. Click **Submit** to continue. If you enter an incorrect password, you will get a message. Click **Close** and restart the logon process.
10. After you have successfully been authenticated, you will see the main panel of the Configuration Client.

4.8.4 Summary of the Configuration Client Main Panel

The parts of the Configuration Client's main panel (see Figure 41 on page 67) are briefly explained below:

Firewall Name This field displays which firewall you are logged into and which administrator ID is being used.

Logon-Logoff The Logoff/Logon button is a reconnect button. When clicked, you are logged off of your current session and the Logon panel is redisplayed. At this point, you can restart the logon sequence to connect to a different firewall or to log on as a different administrator.

Navigation Tree The Navigation Tree, along the left side, provides a means of accessing any of the configuration panels associated with different firewall functions. Different groups of functions have been organized into collapsible *folders*. Double-clicking on the folder icons acts to open or close them. The folders hold *documents*. Double-clicking on a document icon brings up the configuration panel or dialog box for that function. See 4.8.5, "Summary of the Configuration Client Navigation Tree" on page 67 for a brief description of each branch.

Alerts Display This area displays the contents of the local log file. The buttons **Latest** and **Previous** will update the display. Clicking the **Log Viewer** button will bring up the Log Viewer Controls dialog that allows you to browse through your firewall log files.



Figure 41. The Configuration Client's Main Panel

4.8.5 Summary of the Configuration Client Navigation Tree

The main parts of the Navigation Tree of the Configuration Client are briefly explained below:

System Administration Folder The System Administration folder houses the documents for the general configuration of the firewall. The documents it holds are Security Policy, Interfaces, Domain Name Services, Secure Mail Server, and the sub-folder System Logs.

Security Policy The Security Policy document provides a quick and easy way for you to set blanket policies for the Firewall. Most of the check boxes displayed in this window provide a fast path to selecting certain predefined services that will apply to all network traffic received by the Firewall.

Interfaces The Interfaces document displays all network interfaces defined to the machine that the firewall is installed on. This function allows you to tell the IBM Firewall which network interfaces are secure and non-secure.

Domain Name Services Domain Name Services separate the secure and non-secure networks into different name spaces. The goal is to provide full domain name service to hosts inside the secure network while providing minimal information to hosts outside the secure network.

Secure Mail Proxy This document configures Secure Mail Proxy. The mail handler forwards all incoming mail to a centralized mail handler to store and route mail to and from the hosts in the secure network.

System Logs This sub-folder contains additional documents for configuring the logging functions on the firewall.

Users This dialog lets you list the current IBM Firewall users, add and delete users' access to the IBM Firewall, and change the attributes of a user.

Network Objects The Network Objects document allows you to maintain information about network addressable components on your network. This function acts as a central repository for use by other functions in the firewall. Primarily, network objects are used to designate source and destination addresses when you create your connections. Individual network objects can be placed into groups. Use of these groups can save time when configuring connections.

Traffic Control Folder The Traffic Control folder contains dialogs for specifying IP connections that the firewall will handle. The dialogs it holds are Connection Setup, Connection Activation, and the Connection Templates sub-folder.

Connection Setup Connection Setup document brings up the Connection List dialog. From this dialog you can list and add to the protocols and services that the firewall will permit or deny. These protocols and services which are the object of Connection Setup are defined in the Connection Templates sub-folder.

Connection Activation From the Connection Activation document, you regenerate, deactivate, or validate the rules you have defined in Connection Setup. You can also list the current rules as they appear in the raw configuration file, and enable or disable logging of connections.

Connection Template Folder The Connection Template Folder is where you can define templates of service connections that will be used for the Connection Setup. In this folder are dialogs for adding services, rules, and SOCKS rules.

HTTP The HTTP dialog allows you to control the behavior of your HTTP proxy.

Note: In Chapter 6, "Firewall Setup Using the Wizard" on page 109 we use the Wizard to do a similar configuration as the one we will describe in this chapter.

Wizard

In the next sections we will describe how to configure the firewall using the option of the client navigation tree. In Chapter 6, "Firewall Setup Using the Wizard" on page 109 we show how to do a similar job using the firewall Wizard.

4.8.6 Specifying the Secure and Non-Secure Interfaces

Connection privileges to the firewall hinge on the distinction between the secure and non-secure interfaces. Most of the traffic expected in a firewall are outgoing connections from the secure network to the non-secure network; connections in the reverse direction are generally blocked. One of the first steps in configuring the firewall, therefore, is to identify to the system which interface is connected to the trusted network and which interface to the untrusted network.

1. On the System Administration folder of the Navigation Tree, click on the **Interfaces** document to open the corresponding dialog. Here you will see that the firewall has already identified your network adapters by IP address and it designates them as non-secure interfaces.
2. Highlight the adapter which is supposed to be in the secure network. Click on the **Change** button to alter its status.

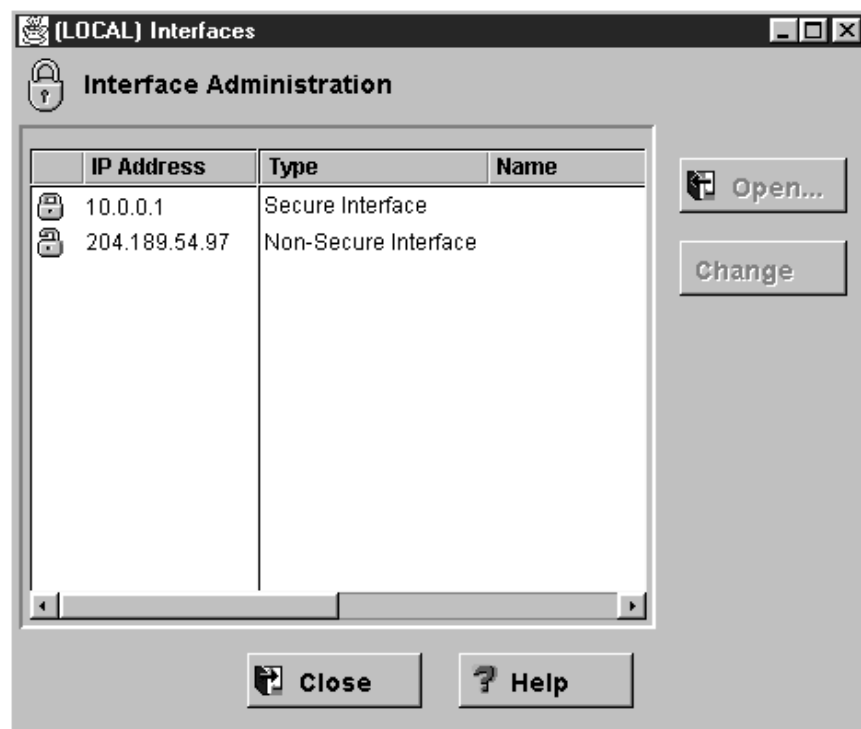


Figure 42. The Interfaces Dialog

3. If you wish, you can give your adapters a significant name. Highlight an adapter and click on the **Open...** button to reveal a dialog with a Name field. Click on **OK** to close the dialog.
4. Click on **Close** to commit the change.

4.8.7 Configuring DNS

Out of the box, IBM eNetwork Firewall for Windows NT expects you to use two name servers, one for the secure network and another for the Internet at large. With this information, IBM eNetwork Firewall for Windows NT will set up a caching DNS server on the firewall system to point to the external DNS. It will also configure the client DNS of the firewall system.

1. On the System Administration folder of the Navigation Tree, click on the **Domain Name Service** document. This dialog begins with blank fields for the secure domain name, the secure DNS server, and the non-secure DNS server.
2. In the Secure Domain Name field, enter the domain name that you will be using in your secure network.

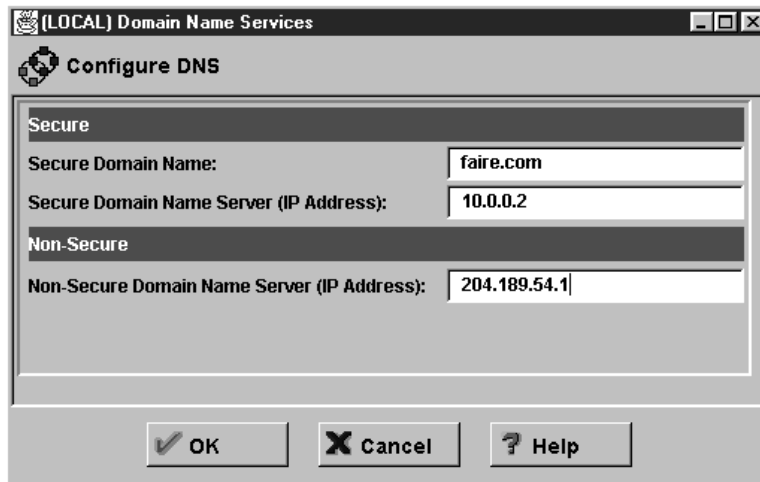


Figure 43. Setting Up the Domain Name on the Firewall

3. In the Secure DNS Server field, enter the IP address of your secure name server.
4. In the Non-Secure Domain Name Server field, enter the IP address of your external name server.

Modifications of the DNS client and server components are performed automatically after you save the information on this dialog.

Configurations not explicitly supported by IBM eNetwork Firewall for Windows NT in the Configuration Client are also possible. See Chapter 11, “Domain Name Service (DNS)” on page 247 for more information.

| 4.8.8 Configuring the Secure Mail Proxy

In the basic configuration of IBM eNetwork Firewall for Windows NT, all incoming mail is received by the firewall and forwarded to one or more internal mail servers. E-mail addresses on all outgoing mail are rewritten so as to hide the internal domain name. Configuring mail for the firewall involves filling in this information.

1. Open the Secure Mail Proxy dialog box of the System Administration folder. In this box you will see no defined hosts, only an entry titled <NEW>.

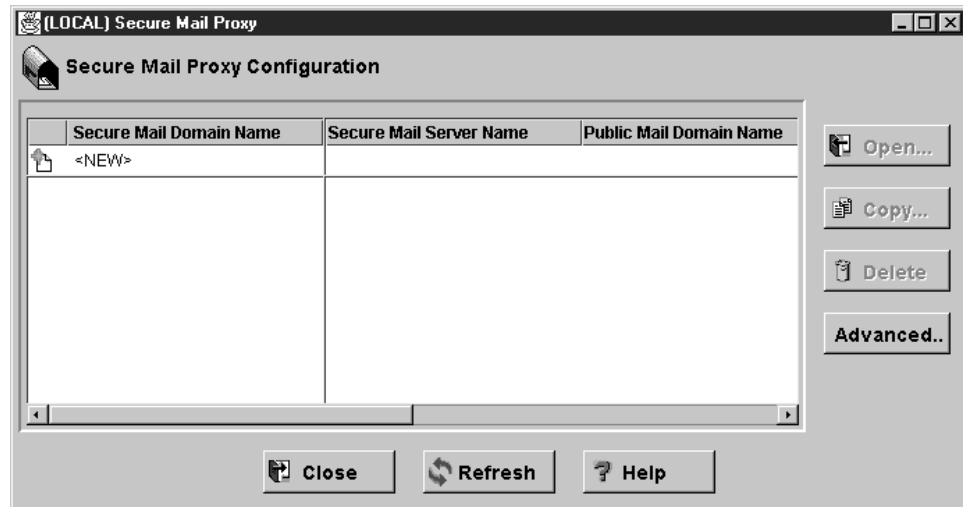


Figure 44. The Secure Mail Proxy Dialog Box

2. Double-click on **<NEW>**. This will bring up a dialog box called Add Mail Server.
3. In the Secure Mail Domain Name field, enter the domain name being used by your internal network.
4. In the Secure Mail Server Name field, enter the host name of your internal mail server.
5. In the Public Mail Domain Name field, enter the domain name to which you want to translate your internal e-mail addresses.

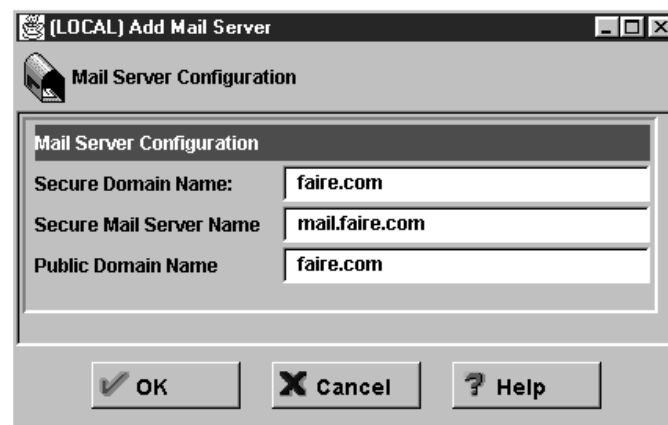


Figure 45. Configuring for Secure Mail Services

Note: The use of the **Advanced...** button of Figure 44 is covered in Chapter 12, “Mail Handling” on page 269.

4.8.9 Creating Network Objects

The firewall needs to know the network objects to which it will apply the security policies. The four basic security objects are the secure network interface, the non-secure network interface, the secure network, and the non-secure network. You can define additional network objects for implementing more sophisticated security connections.

1. Click on the **Network Objects** document on the Navigation Tree. The resulting dialog box shows that only the network object named The World has been defined. The World refers to the Internet at large.

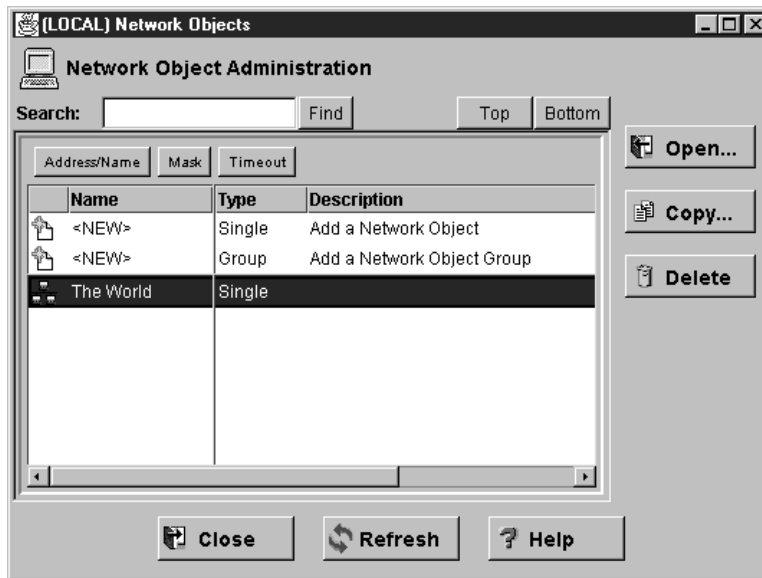


Figure 46. The Network Objects Dialog

2. To create a new Network Object, double-click on the entry **<NEW>** with type Single. This opens the Add a Network Object dialog. The five object types are: Host, Network, Firewall, Interface, and Router.

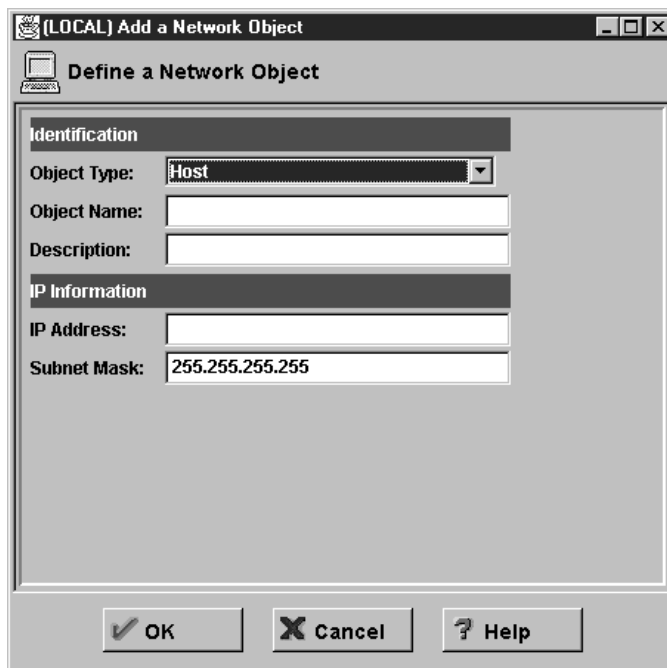


Figure 47. Adding New Network Objects

Not Your Usual Subnet Mask

The Subnet Mask field in this dialog box does not refer to the usual subnet mask that you apply to IP addresses. It is actually a mask that identifies to the firewall which portion of the associated IP address is significant when it applies the filter rules to this network object. However, it does follow the matching rules of the subnet masks that you know.

The simple rule of thumb outlined below should clarify things.

There are five types of network objects: host, network, firewall, router, and interface.

A host is a particular computer on your network, usually a server.

A network is a collective range of network addresses that is characterized by the address mask.

A firewall is a single machine with a firewall installed on it. It refers to another firewall on the network, not the firewall that you are configuring.

A router is what its name suggests.

An interface is a network adapter on a machine. You will use this type to designate the network interfaces on your firewall for the application of the IP filter rules. However, this type is not exclusively limited to adapters on the firewall, but can be adapters on any other machine.

Host, firewall, router, and interface objects should have masks of 255.255.255.255. They are individual objects uniquely identified by their IP address; this means that all the bits of the IP address are significant, and the applied mask indicates just that.

The mask for the network object is the one that bears closest resemblance to subnet masks. As with the other objects, the bits of the mask indicate which bits of the network address are significant, and which ones will be treated as a wildcard. Thus, the mask for a class A network would be 255.0.0.0; for class B, it would be 255.255.0.0; and for class C, 255.255.255.0. Subnets follow similar masking rules.

This information is again discussed in 8.1.2, "Objects and Groups" on page 144.

3. Create a definition for your secure network interface. The Object Type will be Interface. Give it a descriptive name such as Secure Firewall Interface or its fully-qualified domain name. Also fill in the IP Address and the Subnet Mask assigned to the adapter. The Description field is optional. Click on the **OK** button to finish.
4. Create a definition for your non-secure network interface. As with its secure counterpart, the Object Type will be Interface. Give it a descriptive name such as Non-Secure Firewall Interface. Fill in the IP Address and Subnet Mask. The Description field is optional. Click on the **OK** button to finish.
5. Create a definition for the secure network. The Object Type will be Network. As before, give it an obvious name like Secure Network. Enter the network number and its corresponding subnet mask. The Description field is optional. Click on the **OK** button to finish.

6. Click on the **Close** button to commit your changes.

4.8.10 Most Common Settings for Security Policy

It's a wise action to review the settings the firewall's Security Policy dialog before you connect it to the network. There are common options that will apply to most IBM Firewall for NT configurations.

1. Open the **Security Policy** dialog from the System Administration folder. As this will be the first time for you to open it, all the boxes will be unchecked.

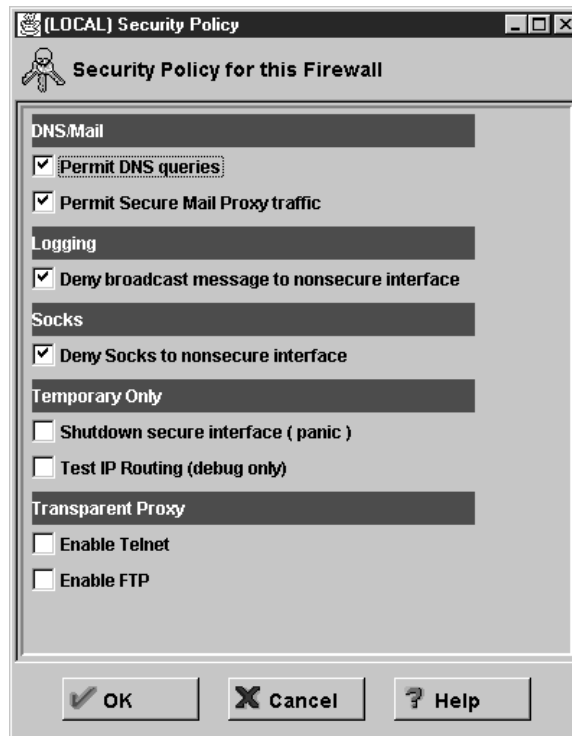


Figure 48. The Security Policy Dialog

2. Check the **Permit DNS queries** box. This will set up filter rules for permitting secure clients to query for DNS names through the firewall. For more restrictive filter settings, see 9.7, "DNS: Domain Name Server" on page 188.
3. Check the **Permit Secure Mail Proxy Traffic** box. This will permit SMTP clients to connect to Secure Mail Proxy. For more restrictive filter settings, see 9.6, "SMTP: Simple Mail Transfer Protocol" on page 184.
4. Check the **Deny broadcast message to non-secure interface** box. This service prevents broadcast messages from being received at the non-secure port. This is useful in reducing the amount of logging especially if your firewall is directly connected to the Internet.
5. Check the **Deny SOCKS to non-secure interface** box. This service prohibits SOCKS traffic coming from the non-secure network to pass through the firewall.
6. Click on the **OK** button. You will see an Information dialog which will ask you if you want to access the Activation panel. Click on the **Yes** button.
7. The Connection Activation dialog comes up. Select the **Regenerate Connection Rules and Activate** radio button. This will enable the changes you've made in the Security Policy dialog.

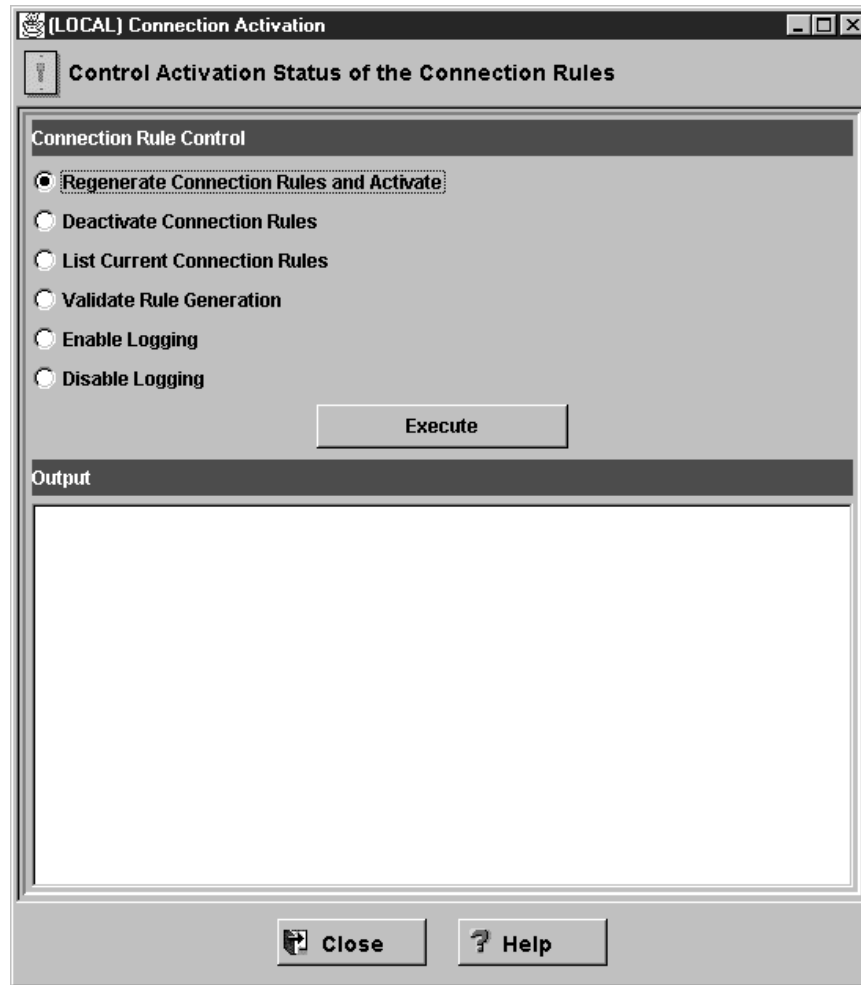


Figure 49. The Connection Activation Dialog

The check boxes in the Temporary Only group should remain unchecked.

Shutdown secure interface (panic) closes the firewall to all traffic. You will want to use this is when you suspect that you are under attack.

Test IP Routing (debug only) opens up the firewall to all traffic. Use this only when you are debugging the firewall. Needless to say, the non-secure interface should only be connected to a controlled network when you do this.

4.9 From Here On...

Our purpose for this chapter has been to walk you through an installation of the IBM eNetwork Firewall for Windows NT, starting from the Windows NT operating system, and ending with a base firewall system. We took a long detour to secure the firewall machine in order to make up for the deficiencies in the default Windows NT installation and to extend the hardening process done by the IBM Firewall. Having completed all the steps outlined in this chapter, the firewall is now ready for further customization.

- To get started with a basic configuration that meets the Internet security needs of small- and medium-sized companies, see Chapter 5, “A Quick and Basic Configuration: A Case Study” on page 77.
- To learn how to get a basic firewall configuration up and running quickly in using the new wizard see Chapter 6, “Firewall Setup Using the Wizard” on page 109.
- To learn how to customize the firewall's IP filters, see Chapter 8, “IBM Firewall Rule Base” on page 143. For examples of IP filters, see Chapter 9, “Examples of Rules for Specific Services” on page 161.
- To learn how to configure proxy services, see Chapter 13, “Configuring Proxy Services and SOCKS” on page 293.
- To learn how to configure a Virtual Private Network, see Chapter 15, “Virtual Private Network” on page 357.

Chapter 5. A Quick and Basic Configuration: A Case Study

In this chapter, we will take you through a quick and basic configuration of the IBM Firewall for NT in a typical intranet/Internet environment. To provide a basis for concrete points of discussion, we will configure the firewall in a case study involving a hypothetical company. For this same case study, we will also look at the setup of peripheral Internet services such as DNS servers and mail servers in the context of the firewall.

5.1 Case Study: Faire.Com

Faire.Com is a small computer consulting company. Their primary line of business is Lotus Notes programming and application development within the Windows NT environment. A significant portion of the business relies on the Internet: consultants correspond with customers via e-mail, and programmers research new developments on the Web.

The Faire.Com intranet is composed of three Notes/Domino development servers, a Notes mail server, and 40 client workstations. Apart from Notes mail and databases, the Notes mail server also acts as an SMTP gateway for Faire.Com employees to send to and receive e-mail from the Internet. The servers all run on Windows NT Server 4.0 and the clients run on Windows NT Workstation 4.0.

Faire.Com's needs with regard to the Internet are well-defined. Faire.Com consultants and programmers basically enjoy unrestricted access to e-mail and the World Wide Web but they use no other services. As for access from the public Internet: except for e-mail, no one outside the company intranet will be permitted in, least of all to the three development servers, and Faire.Com's management would like to keep it that way.

Faire.Com's Internet connection is via a local Internet Service Provider (ISP). In order to save on monthly connectivity expenses, Faire.Com's management opted for a dedicated leased-line connection but with a subnetted portion of the ISP's network. Faire.Com thus only has six public IP addresses assigned to it. Faire.Com uses two of these IP addresses for two Windows NT servers and another IP address for the router.

For its domain name, Faire.Com registered faire.com. Management has decided to use the same domain name for both the public Internet and for the company's private intranet. The public primary DNS server for faire.com. is hosted at the ISP's site.

Faire.Com purchased IBM eNetwork Firewall for Windows NT to fulfill a two-fold function. First, to insulate their intranet servers from the public Internet; and second, to save on cost by creating a private network that allows their 50 or so employees to unlimited access the Internet.

5.2 Planning

The configuration of the firewall will depend on the environment that it is going to function in. Considerations to take into account would be Faire.Com's proposed network connections, security policies, mail services, and domain name services.

5.2.1 Faire.Com's Intranet/Internet Layout

Faire.Com's network layout is shown below.

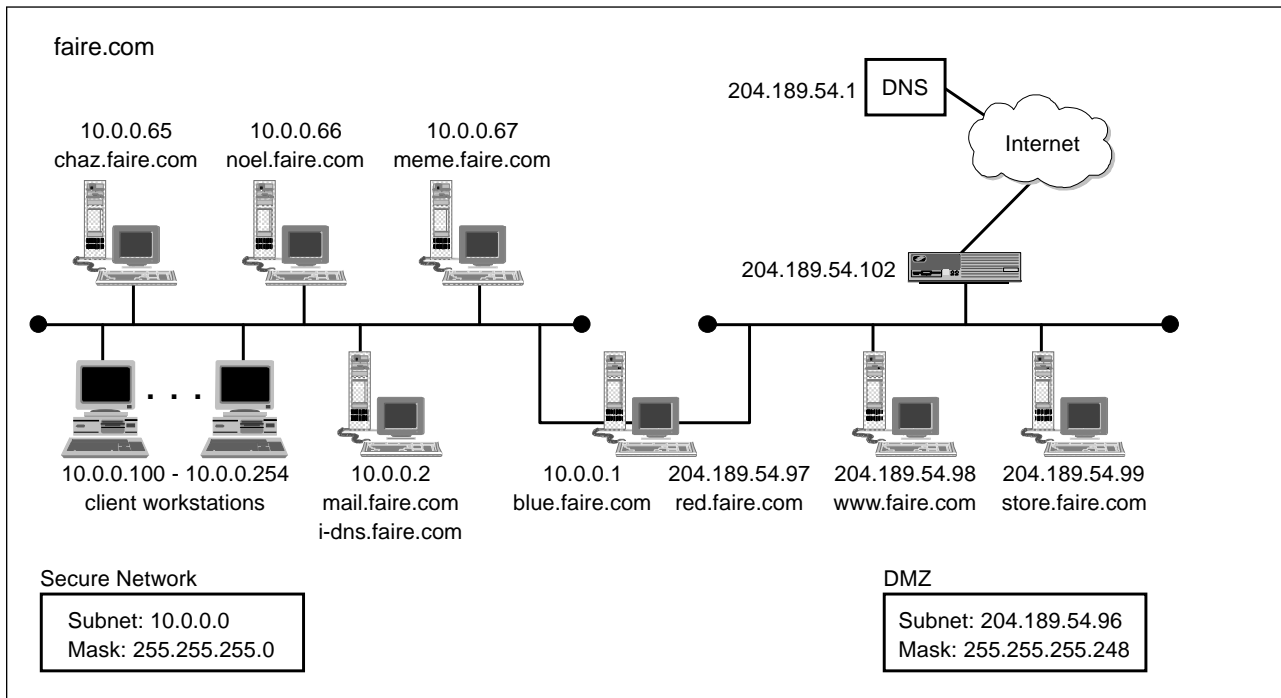


Figure 50. Faire.Com's Network Layout

The firewall is the dividing point between secure subnet and a demilitarized zone (DMZ).

The designated DMZ is a subnet that sits between the firewall and the router going to Faire.Com's ISP. The servers in this network are publicly accessible. On this DMZ are the two Windows NT Internet servers which act as Faire.Com's public Web servers, www.faire.com and store.faire.com. Four public IP addresses are in use here: two for the Web servers, one for the router, and one for the nonsecure interface of the firewall.

The secure subnet consists of three development servers, one mail server, and 40 client workstations. The servers are run on Windows NT Server 4.0 and the clients on Windows NT Workstation 4.0. To support this number of computers on the secure subnet, Faire.Com uses RFC 1627 IP addresses. RFC 1627 assigns IP addresses for use exclusively on private networks.

RFC 1627

RFC 1627 reserves the following network numbers for exclusive use on firewalls and private networks.

- Class A network 10
- Class B networks 172.16 to 172.31
- Class C networks 192.168.0 to 192.168.255

Given this network layout and the requirements of the company, how would you go about configuring the firewall and the other systems?

5.2.2 Internet Security Policy

Given its simple needs, Faire.Com's Internet security policy enforcement by the firewall will be straightforward. Below is a list of activities that the firewall will permit, and their corresponding technical implementations. As the IBM Firewall for NT disables all services by default, you will notice that the actions required in a first-time setup are invariably concerned with enabling traffic. However, you must take care that none of the other services you enable inadvertently conflict with each other.

The planning tables presented in Chapter 3, "Planning Your Firewall Installation" on page 29 are based on the Faire.Com case study. Refer back to 3.2, "Defining Network Objects" on page 31 and the subsequent planning tables for ways in which to create planning sheets for the Internet security policy.

5.2.2.1 Inbound Services

Policy: Incoming e-mail to be forwarded to the internal mail server; no restrictions or monitoring of messages.

Implementation: Set up an internal mail server. Enable Secure Mail Proxy on the firewall and specify the internal mail server to which all messages will be forwarded to.

5.2.2.2 Outbound Services

Policy: Outgoing e-mail; no restrictions or monitoring of messages.

Implementation: Enable Secure Mail Proxy on the firewall.

Policy: Web browsing

Implementation: Enable SOCKS on the firewall, and permit HTTP and passive FTP services. FTP is required because Web clients must sometimes download files from FTP servers; passive FTP is a mode of operation which is more secure than normal FTP.

Configure the Web clients to make use of SOCKS.

Normal and Passive Mode FTP

Two ports are involved in FTP: one for a command channel, and another for a data channel. The difference between normal mode and passive mode lies in their use of the data channel. With normal mode FTP, the server opens the data channel of the client, a potentially nonsecure mode of operation. With passive mode FTP, it is the client which opens the data channel to the server.

Both FTP client and server must support passive mode for it to work. Fortunately, passive mode is quite popular and almost all implementations of FTP now incorporate it.

5.2.2.3 Naming Services

Policy: Internal host names will be hidden, but Faire.Com users must still be able to resolve all host names in the public Internet.

Implementation: Set up an internal name server and configure the firewall for hidden DNS.

5.2.2.4 Firewall Administrators

Policy: There are three designated firewall administrators for Faire.Com. They both manage the firewall locally, using the primary administrator account.

Implementation: No action required. Do not enable remote administration.

5.2.2.5 Logging Alerts

Policy: Alert the administrators of any changes made in the firewall.

Policy: Alert the administrators of performance-related events. These would refer to errors in the SOCKS proxy or in firewall connections

5.2.2.6 Archiving

Policy: Archive files daily, and keep them for one week.

5.2.3 Network Details

Using the configuration sheet in 3.12, "IBM eNetwork Firewall for Windows NT Initial Configuration Parameters" on page 41, the details of the firewall for Faire.Com are as follows.

Table 10. Faire.Com's Network Details

Firewall Name	NTFW
Secure Network's Network Number	10.0.0.0
Network Subnet Mask	255.255.255.0
Secure Network Interface Name	blue.faire.com
Secure Network Interface IP Address	10.0.0.1
Secure Network Subnet Mask	255.255.255.0
Non-secure Network Interface Name	red.faire.com
Non-secure Network Interface IP Address	204.189.54.97
Non-secure Network Subnet Mask	255.255.255.248
Internet Gateway IP Address	204.189.54.102

Secure Domain Name	faire.com
Public Domain Name	faire.com
Secure Domain Name Server IP Address	10.0.0.2
Public Domain Name Server IP Address	204.189.54.1
Internal Mail Server Host Name	mail.faire.com

5.3 Configuring Peripheral Services

In relation to the firewall, there are two essential peripheral services in a typical installation: mail and DNS services. DNS and mail services are dealt with more extensively in Chapter 11, “Domain Name Service (DNS)” on page 247 and Chapter 12, “Mail Handling” on page 269.

5.3.1 Configuring the External Name Server

The ISP hosts the primary name server for Faire.Com's domain name, along with the domain names of the other companies that it is servicing. For faire.com, the ISP needs the following entries (other DNS file information have been omitted for clarity):

In the zone file for faire.com:

faire.com	IN NS	dns1.isp.com
*.faire.com	IN MX 10	red.faire.com
	IN MX 20	mail.isp.com
red.faire.com	IN A	204.189.54.97
www.faire.com	IN A	204.189.54.98
store.faire.com	IN A	204.189.54.99

dns1.isp.com is the name server of the ISP. The NS line in the zone file means that it is authoritative for faire.com. Its IP address, in this case study, is given as 204.189.54.1.

The first MX record means that all mail addressed to faire.com must be forwarded to red.faire.com, the non-secure interface of the firewall. The second MX record means that if red.faire.com is for some reason unavailable, mail will be forwarded to mail.isp.com, the ISP's mail server, where it will be stored until red.faire.com comes back on line.

In the reverse mapping zone file for 54.189.204.in-addr.arpa,

97	IN PTR	red.faire.com.
98	IN PTR	www.faire.com.
99	IN PTR	store.faire.com.

For cases when the company must maintain its own DNS, please refer to Chapter 11, “Domain Name Service (DNS)” on page 247 for the possible configurations.

5.3.2 Configuring the Internal Name Server

mail.faire.com, Faire.Com's mail server in the secure network, will double as the company's internal name server. This task of setting this up is divided into four parts:

- Install and set up a primary DNS server
- Create zone files for the internal domain faire.com and its reverse name map
- Set up the address, name server, and mail exchanger records
- Configure the internal name server to query the firewall DNS cache for any records that it cannot resolve

To install and set up the primary name server for Faire.Com:

1. Install the Microsoft DNS Server on mail.faire.com. Follow the steps given in 4.4.3, "Installing the Microsoft DNS Server" on page 49 to do this.
2. Click on **Start->Administrative Tools (Common)->DNS Manager** to bring up the DNS Manager.



Figure 51. The DNS Manager

3. On the DNS Manager, highlight the Server List object in the left panel. Click on **DNS** from the pull-down menu and select **New Server** to create the instance for the DNS server.

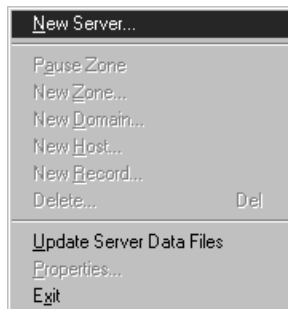


Figure 52. Adding a New Server from the DNS Manager

- Enter the IP address of mail.faire.com from the Add DNS Server dialog box. Click on **OK**. A branch named 10.0.0.2 (mail.faire.com's IP address) will appear under the Server List group.

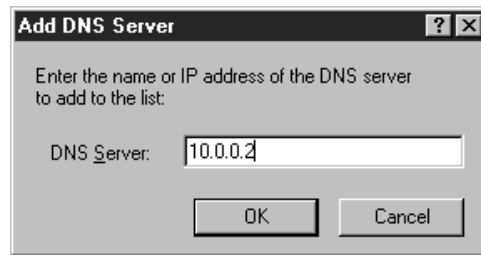


Figure 53. Adding a New Server from the DNS Manager

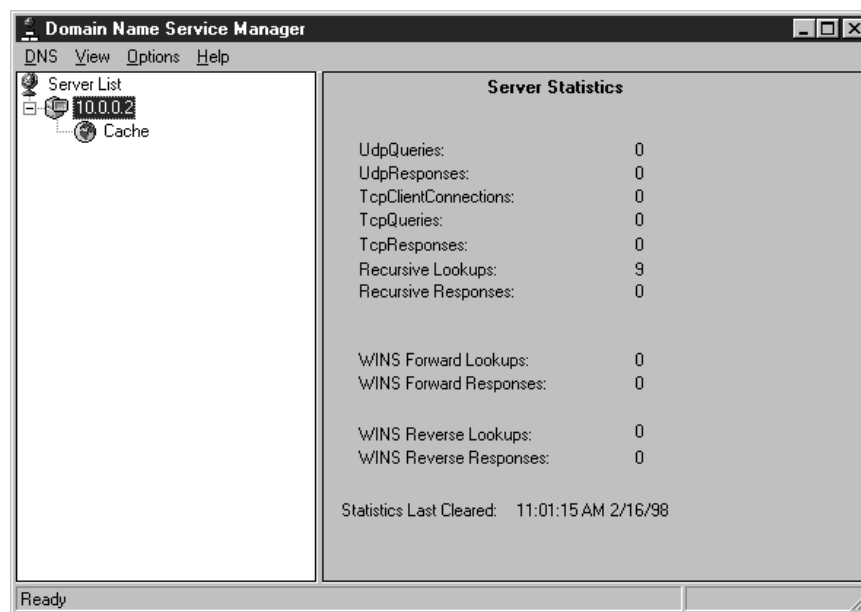


Figure 54. A New DNS Server

To set up the zone files:

- Highlight the entry 10.0.0.2. Open the **DNS** pull-down menu and click on **New Zone**.

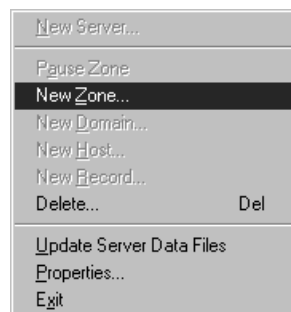


Figure 55. Adding a New Zone

2. A dialog titled Creating new zone for 10.0.0.2 should appear. Select the **Primary** radio button to create a primary zone. Click on **Next** to go to the next screen.

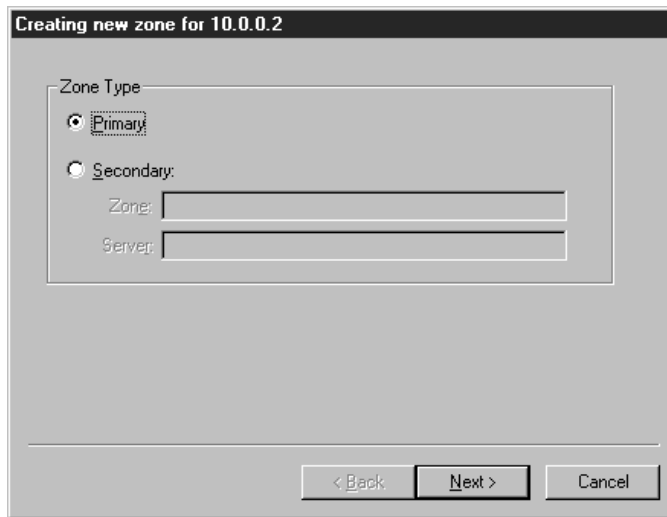


Figure 56. Creating a New Zone

3. You will be prompted to enter the details of the zone. In the Zone Name field, type in faire.com. As you click on the Zone File field, the entry faire.com.dns should automatically appear. You can change the zone file name or you can take it as it is. Click on **Next** to continue.

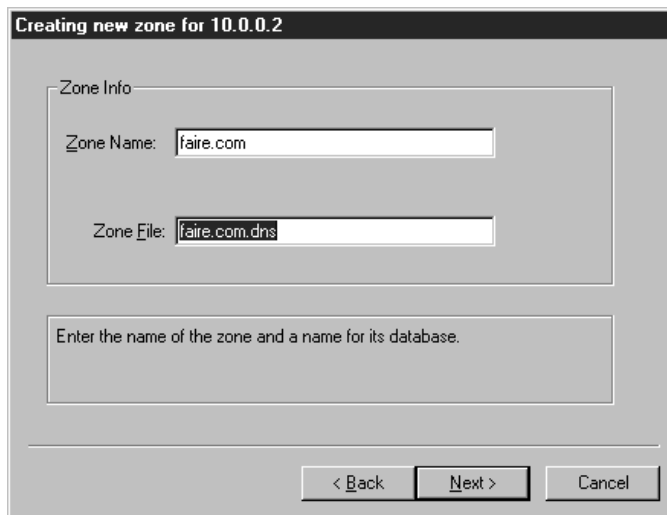


Figure 57. New Zone Name and Its Associated DNS File

4. A confirmation screen appears. Click on **Finish** to create the zone. On the main DNS Manager screen, the zone faire.com now appears as a branch on the Server List tree. Note the entries on the right panel called Zone Info: the name server (NS) and start-of-authority (SOA) are automatically created.

Note: It's actually a good idea to use a separate name, such as i-dns.faire.com, to allow the functions to be easily moved to different servers in the future. But for simplicity we use the single name mail.faire.com here.

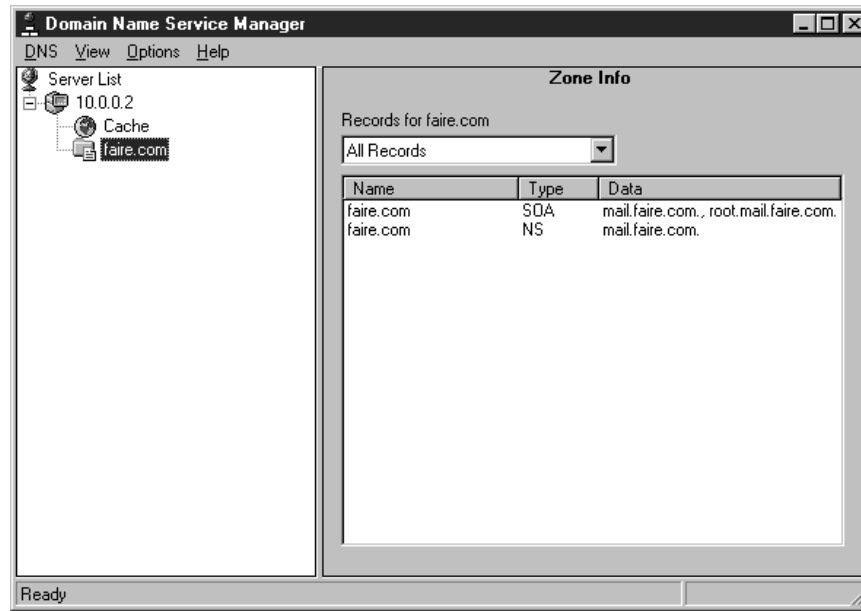


Figure 58. The New Zone and Some Predefined Entries

5. Next, create the reverse map. On the Server List tree, highlight the entry 10.0.0.2 once more. As before, open DNS from the pull-down menu and click on New Zone.
6. On the Zone-creation dialog, select the **Primary** radio button. Click **Next**.
7. In the Zone details, type in 0.0.10.in-addr.arpa. As you click or tab to the Zone File field, the entry 0.0.10.in-addr.arpa.dns should automatically appear. You can change the zone file name or you can take it as it is. Click on **Next** to continue.
8. The confirmation screen appears. Click on **Finish** to create this reverse-name zone. On the main DNS Manager screen, the zone 0.0.10.in-addr.arpa now appears as a branch on the Server List tree. NS and SOA information are also automatically created, and can be seen on the right panel list.

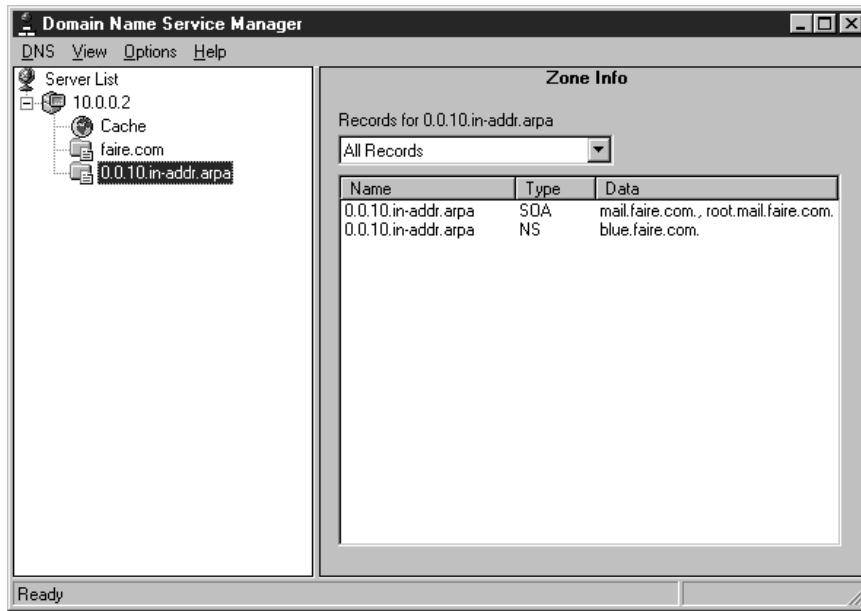


Figure 59. The Reverse Name Mapping Zone

To create records on the name server:

1. Highlight the **faire.com** branch of the Server List tree of the DNS Manager by clicking on it.
2. From the DNS pull-down menu, click on **New Host**. A New Host dialog appears.

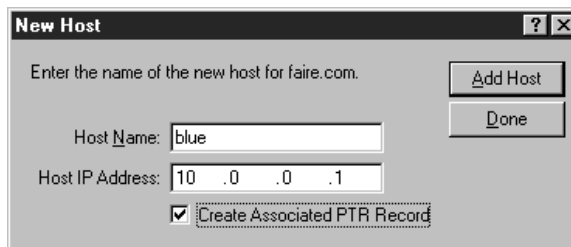


Figure 60. Adding New Host Information.

3. On the New Host dialog, you fill in the Host Name and Host IP Addresses fields to create new records for the faire.com domain. Click on the **Add Host** button to add a new entry into the DNS database.
4. Create entries for the hosts in the secure network. As you add these entries in the New Host dialog, make sure the Create Associated PTR Record box is checked. This will generate PTR records in the 0.0.10.in-addr.arpa zone file automatically.

Host Name	Host IP Address
blue	10.0.0.1
mail	10.0.0.2
chaz	10.0.0.65
noel	10.0.0.66
meme	10.0.0.67

5. Create entries for faire.com hosts in the DMZ. Uncheck the **Create Associated PTR Record** box, because these fall within the authority of Faire.Com's ISP, and Faire.Com does not control the 54.189.204.in-addr.arpa zone.

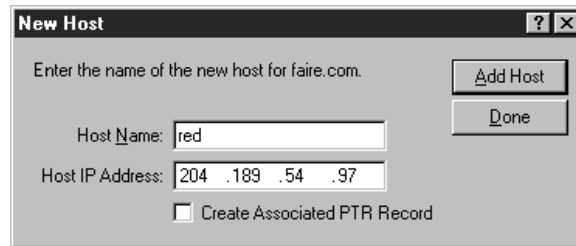


Figure 61. Adding New Host Information

Host Name	Host IP Address
red	204.189.54.97
www	204.189.54.98
store	204.189.54.99

6. Click on **Done** of the New Host dialog to close it.
7. If you click on the 0.0.10.in-addr.arpa branch of the Server List tree, you might note that your PTR records for your hosts have not been added. Do not be alarmed; they are actually there. From the View pull-down menu of the DNS Manager, click on **Refresh**, or simply press F5. The Zone Info on the right panel will re-read the files and show that your reverse-map records have indeed been added.

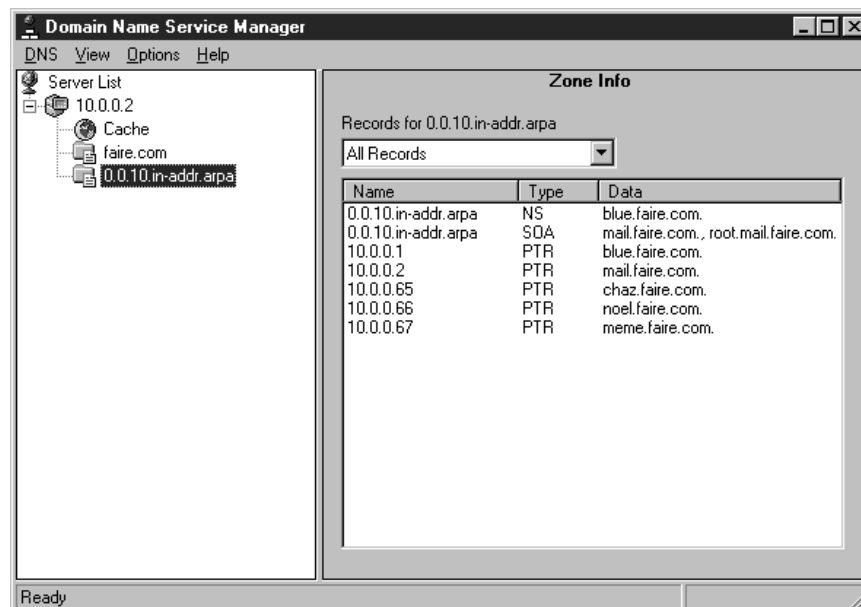


Figure 62. Reverse Name Mapping Information

Inside DNS

Despite the graphical interface, the Microsoft DNS Server still uses traditional DNS files for its name-to-address databases. Having gone through the exercise above, you will note that the files `faire.com.dns` and `0.0.10.in-addr.arpa` have been created in the `c:\WINNT\system32\DNS` directory. Please see Chapter 11, "Domain Name Service (DNS)" on page 247 for more information.

An Optional MX Record for Internal Users

If users in `Faire.Com` will send mail to each other using an SMTP mail client, and they want to use the address format `user@faire.com`, then you will need to create a record for an internal mail exchange in the DNS server. To do this:

1. Click on the **faire.com** branch of the Server List tree. From the DNS pull-down menu, click on **New Record...** You will get the New Resource Record dialog.
2. From the Record Type list, locate and click on **MX Record**. In the Mail Exchange Server DNS Name field, type in `mail`. Type in any number for Preference Number. Leave the Host Name (Optional) field blank. Click on **OK** to close the dialog.

To tell the name server to query the firewall DNS cache for records it cannot resolve,

1. Click on the branch `10.0.0.2` of the Server List tree. From the DNS pull-down menu, click on **Properties...** The Server Properties dialog will appear.
2. Click on the **Forwarders** tab. The dialog will show a Use Forwarder(s) checkbox, with some other elements grayed out.
3. Check the **Use Forwarder(s)** box. The other elements become active.
4. Check on the **Operate As Slave Server** box. This will tell the name server to query only the designated forwarders for name resolution and no other.
5. Enter the address of the firewall's secure adapter in the IP address field. Click on **Add** to place it in the list.
6. Increase the Forward Time Out value to 5 seconds.

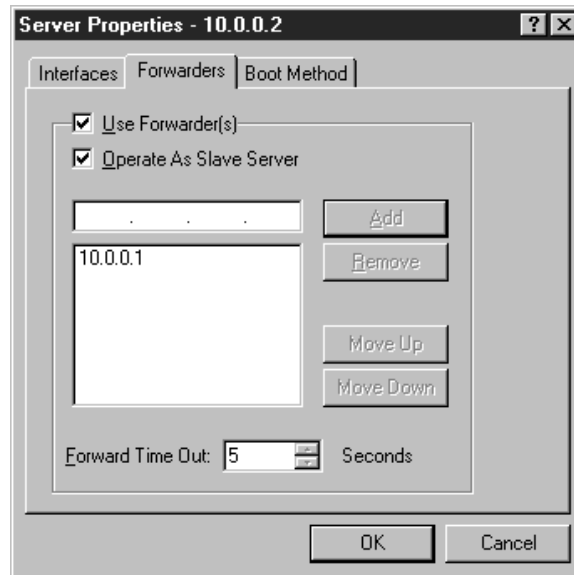


Figure 63. Adding a Forwarders Entry

5.3.3 Configuring the Internal Mail Server

The internal mail server needs only one minor modification: it must relay all outgoing SMTP messages to the secure adapter of the firewall, blue.faire.com

In this case study, Faire.Com uses a Lotus Notes/Domino 4.5 server with the SMTP MTA 1.5 loaded on it. To configure the relay for this example (applicable only to Domino 4.5 or higher):

1. With the Lotus Domino Server running in the background, open the Lotus Notes client on the server machine.
2. From the Administrator workspace, double-click on the icon for the database called Faire.Com's Address Book on Local.
3. On the left panel of the open database you will see the tree for Folders and Views. Expand this tree, and you will see a subtree called Server.
4. Open the **Server** subtree and click on **Connections**. The right panel should reveal database entries for various connections maintained by the Domino server, including an SMTP connection.
5. Double-click on the line of type SMTP. This will open the Notes document with the details of the configuration of the server's SMTP connections.
6. Click on the **Edit Connection** button.
7. Locate the field Destination server. This should be under the Basics group, the very first group of the document. On this field, enter the host name of the mail relay, in this case, the firewall. Type in blue.faire.com.
8. In the Optional network address field, type in the IP address of faire.com. This would be 10.0.0.1.
9. Click on **Close** to save the document.

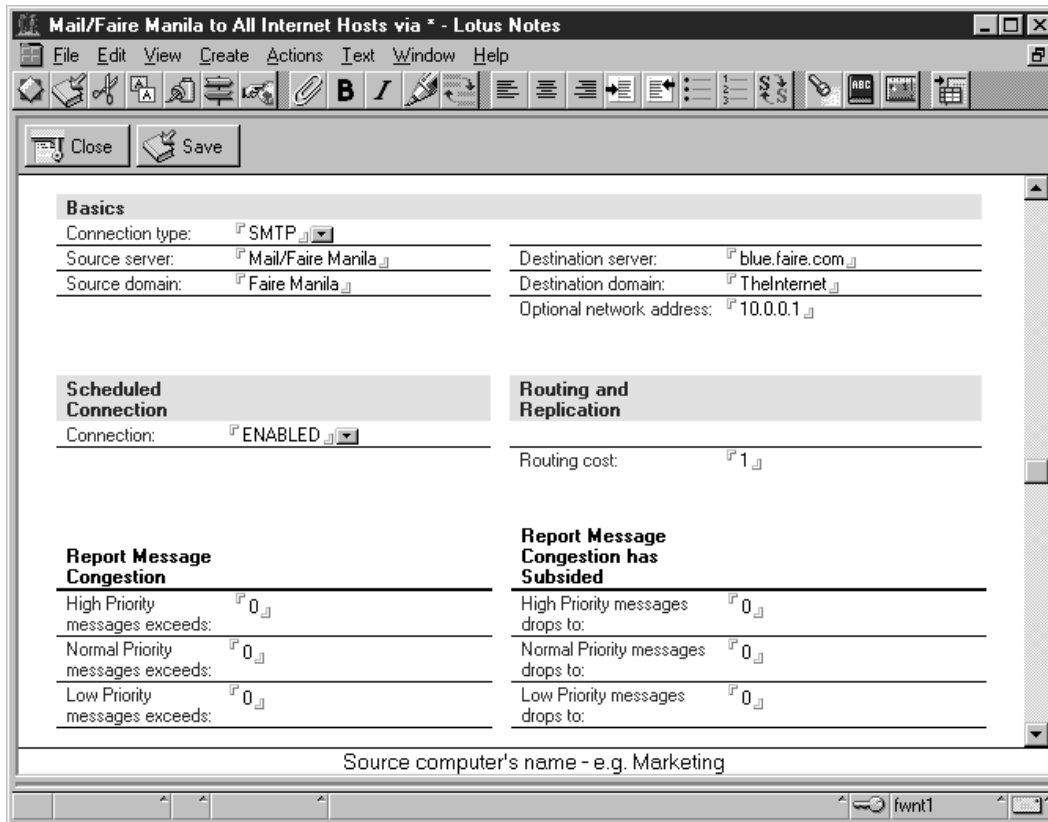


Figure 64. Configuring Lotus Notes SMTP MTA to Forward Mail to the Firewall

In Real Life

If Faire.Com were using some other SMTP server, the relay principle would still hold, though the configuration steps will be different. Consult your mail server's manual to learn how to configure mail relays.

5.3.4 DNS Client Configuration

All Windows NT servers and workstations in the secure network must use the internal name server for DNS resolution. Configure the DNS client on each internal machine (including the secure name server). To do this:

1. Open the Network object from **Start**→**Settings**→**Control Panel**.
2. Click on the **Protocols** tab. Highlight **TCP/IP Protocol** from the list and click on **Properties**. This will show the Microsoft TCP/IP Properties dialog.
3. Click on the **DNS** tab. The fields you see here will configure the DNS client of the system.
4. Enter the host name of the machine you are configuring in the Host Name field. In the Domain field, type in faire.com.
5. In the DNS Service Search Order subgroup, click on **Add...**
6. In the resulting dialog box, type in the IP address of the internal name server. In this case, it will be 10.0.0.2. Click on **Add** to finish.
7. In the Domain Suffix Search Order subgroup, click on **Add...**

8. In the resulting dialog box, type in the domain name `faire.com`. Click on **Add** to finish.

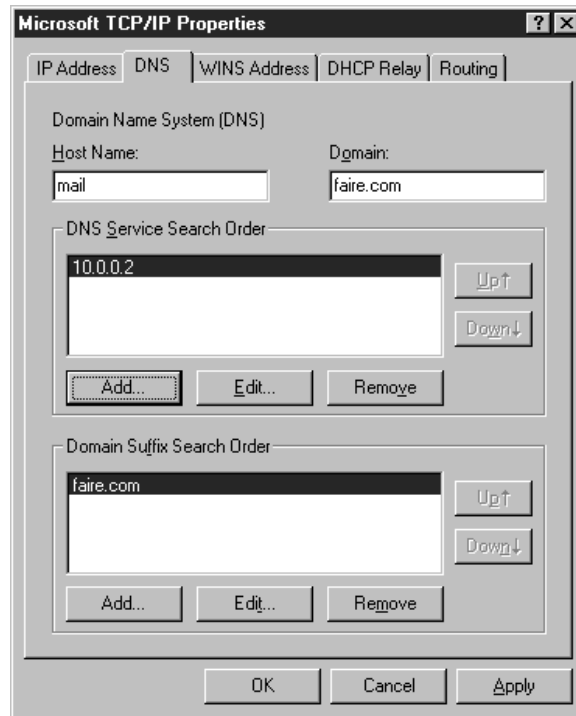


Figure 65. Configuring the DNS Client Component

5.3.5 Default Gateway Configuration

All servers and workstations in the secure network must use the firewall as their default gateway to the Internet. To configure this, perform the following steps on each and every server on the internal network (except the firewall):

1. Open the **Network Control Panel**.
2. Click on the **Protocols** tab. Highlight **TCP/IP Protocol** from the list and click on **Properties**. This will show the Microsoft TCP/IP Properties dialog.
3. Click on the **IP Address** tab. In the Default Gateway field, enter the IP address of the secure adapter of the firewall, `blue.faire.com`. This would be `10.0.0.1`.

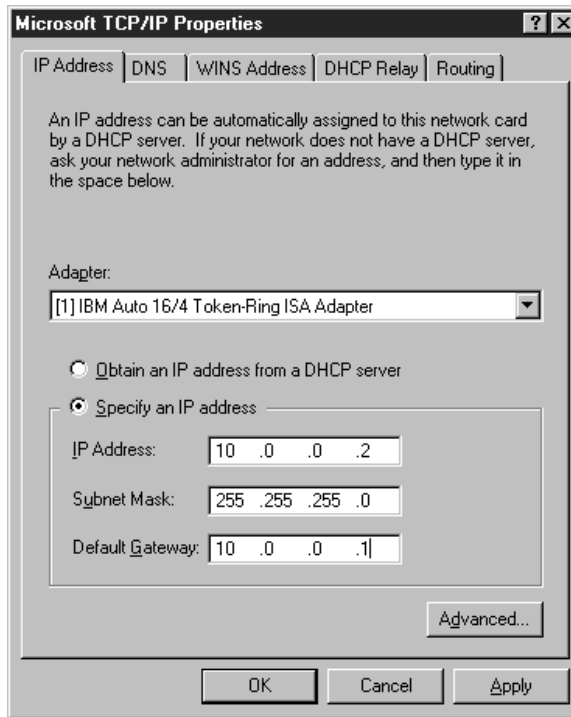


Figure 66. Configuring the Default Gateway

4. Click on **OK** to confirm. Reboot the system.

5.4 Configuring IBM eNetwork Firewall for Windows NT

With the DNS and mail servers properly configured for operation, we now proceed to the configuration of the firewall itself. Configure the firewall's network interfaces using the information in 5.2, "Planning" on page 78

In the following steps, we will assume a freshly installed IBM Firewall for NT. Start the Configuration Client and log on to begin.

5.4.1 Configuring the Interfaces

Follow these steps to configure the Interfaces:

1. On the System Administration folder of the Navigation Tree, click on the **Interfaces** document to open the corresponding dialog. The two adapters, 10.0.0.1 and 204.189.54.97 should be in the list, and both should be marked Non-Secure Interface.
2. Highlight the adapter that is supposed to be in the secure network, in this case 10.0.0.1. Click on the **Change** button to alter its type to Secure Interface.
3. Click on **Close** to commit the change.

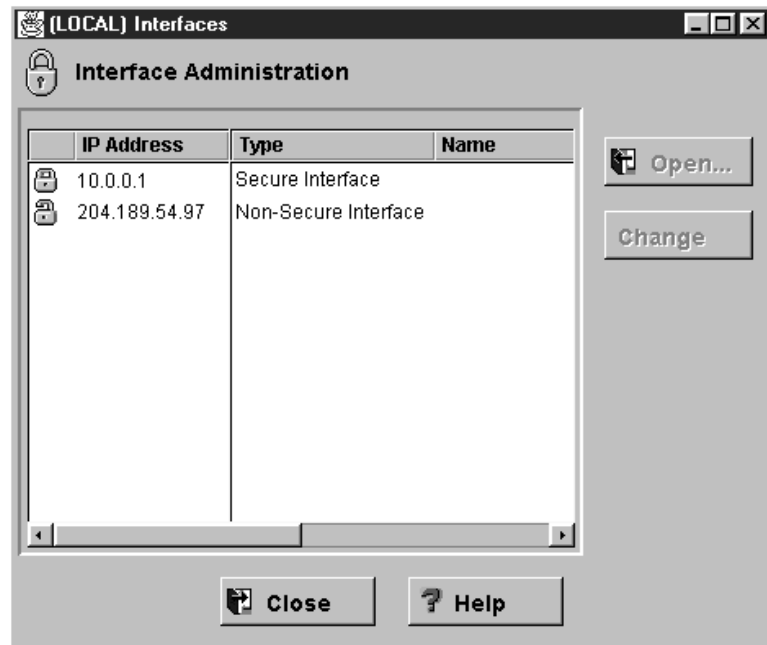


Figure 67. Configuring the Secure and Non-Secure Interfaces

5.4.2 Configuring Name Services

Follow these steps to configure the Name Services:

1. On the System Administration folder of the Navigation Tree, click on the **Domain Name Service** document. This dialog begins with blank fields for the secure domain name, the secure DNS server, and the non-secure DNS server.
2. In the Secure Domain Name field, enter `faire.com`.
3. In the Secure DNS Server field, enter the IP address of the secure name server, `10.0.0.2`.
4. In the Non-Secure Domain Name Server field, enter the IP address of your external name server, in this case, `204.189.54.1`, the name server of the ISP. Click on **OK** to commit the change.

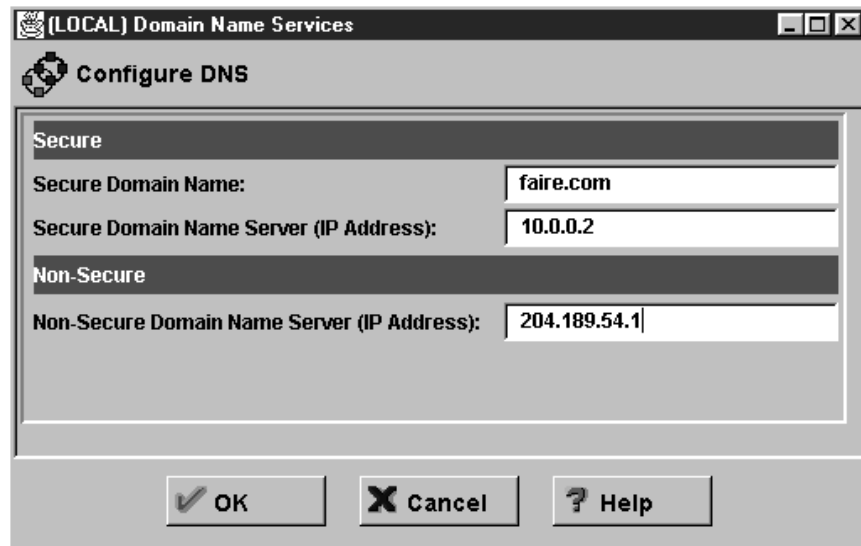


Figure 68. Configuring DNS on the Firewall

5.4.3 Configuring for Secure Mail Services

Follow these steps to configure the Secure Mail Services:

1. Open the **Secure Mail Proxy** document of the System Administration folder.
2. Double-click on **<NEW>**. This will bring up a dialog box called Add Mail Server.
3. In the Secure Domain Name field, enter the domain name being used by your internal network, faire.com.
4. In the Secure Mail Server Name field, enter the host name of your internal mail server. In this case, it would be mail.faire.com.
5. In the Public Domain Name field, enter the domain name to which you want to translate your internal e-mail addresses into. In this case, it would be faire.com. Click **OK** to commit the changes.

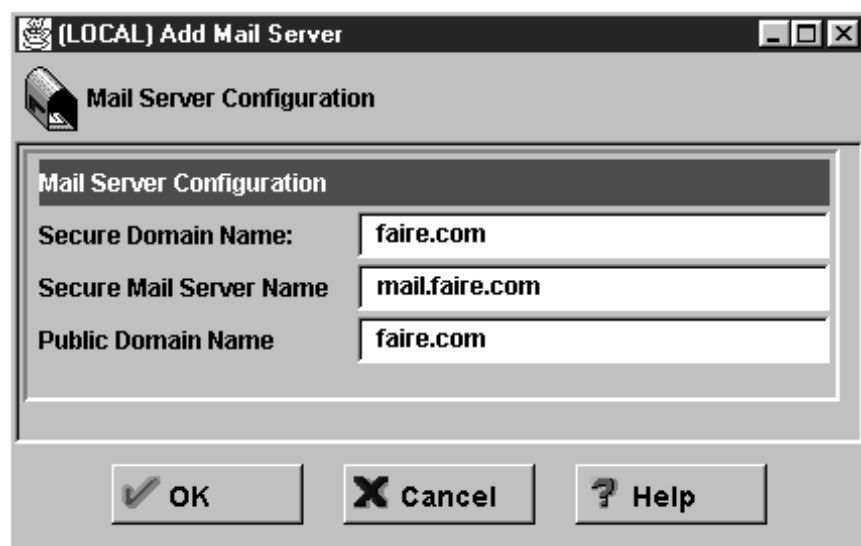


Figure 69. Configuring for Secure Mail Services

5.4.4 Creating Network Objects

Follow these steps to create the Network Objects:

1. Click on the **Network Objects** document on the Navigation Tree. The resulting dialog box shows that only the network object named The World has been defined.
2. Create a new network object by double-clicking on the entry **<NEW>** with type Single.
3. Create a definition for your secure network interface. The Object Type will be Interface. Name it as the Secure Firewall Interface. Under description, you might want to type blue.faire.com. Fill in the IP Address and the Subnet Mask assigned to the adapter. These would be 10.0.0.1 and 255.255.255.255, respectively. Click on the **OK** button to finish.
4. Create a definition for your non-secure network interface. As with its secure counterpart, the Object Type will be Interface. Name it as the Non-Secure Firewall Interface and give it a description of red.faire.com. Fill in the IP Address and Subnet Mask, 204.189.54.97 and 255.255.255.255, respectively. Click on the **OK** button to finish.
5. Create a definition for the secure network. The Object Type will be Network. As before, give it a name such as Secure Network. Enter the network number and its corresponding filter mask. These would be 10.0.0.0 and 255.255.255.0, respectively. Click on the **OK** button to finish.

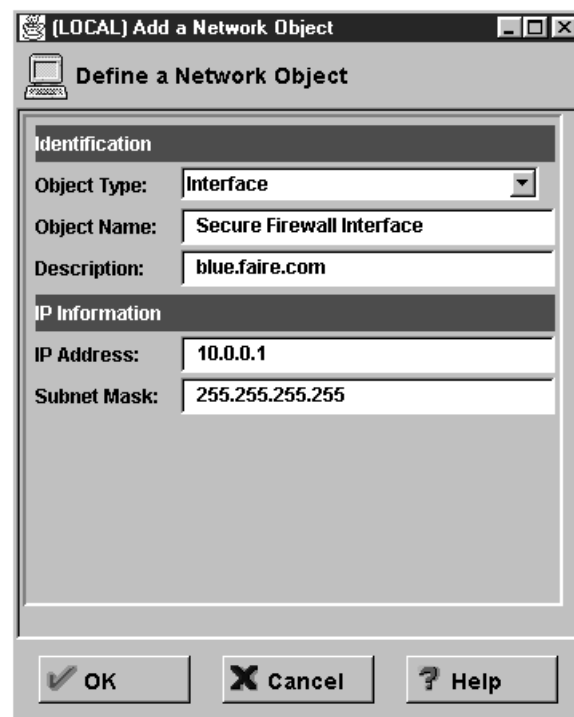


Figure 70. Creating Network Objects for Faire.Com

5.4.5 Configuring Security Policies

Follow these steps to configure the Security Policies:

1. Open the **Security Policy** dialog from the **System Administration** folder.
2. Check the **Permit DNS queries** box.
3. Check the **Permit Secure Mail Proxy traffic** box.

Stricter Security

9.6, "SMTP: Simple Mail Transfer Protocol" on page 184 and 9.7, "DNS: Domain Name Server" on page 188 define more secure filter rules than would normally be given by the firewall under the Security Policy.

Refer to these two sections if you want to implement tighter security for your firewall's DNS and mail services.

4. Check the **Deny broadcast message to non-secure interface** box.
5. Check the **Deny SOCKS to non-secure interface** box.

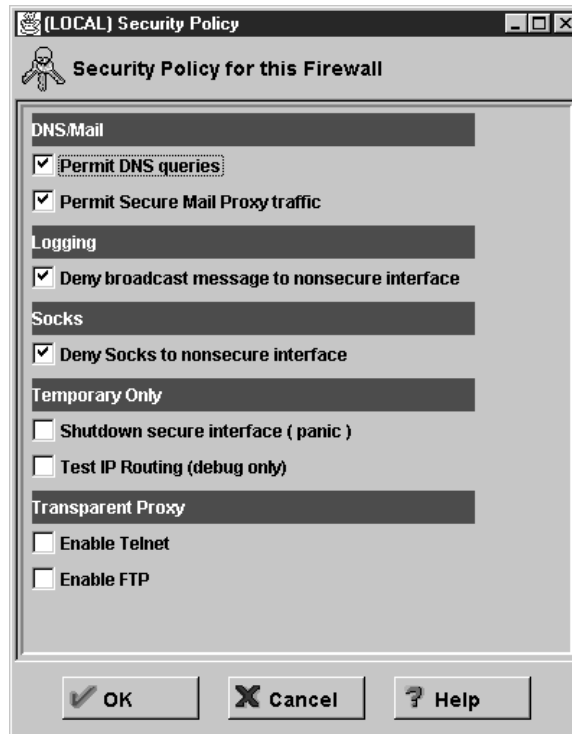


Figure 71. Defining the General Security Policy

6. Click on the **OK** button. You will see an Information dialog which will ask you if you want to access the Activation panel. Click on the **Yes** button.
7. The Connection Activation dialog comes up. Select the **Regenerate Connection Rules and Activate** radio button and click on **Execute**. This will enable the changes you've made in the Security Policy dialog.

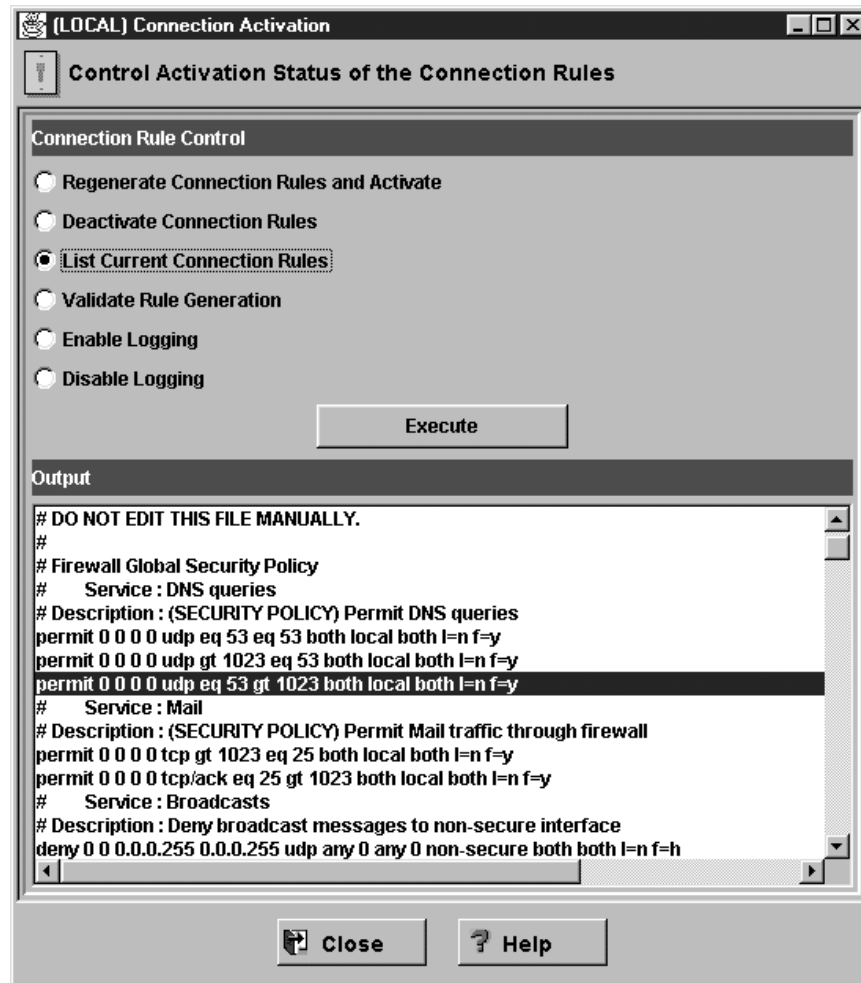


Figure 72. The Connection Activation Dialog

5.4.6 Configuring SOCKS Services

Setting up SOCKS services is discussed in great detail in 13.7, “Using the SOCKS Server” on page 324. For basic configuration for the situation in Faire.Com, follow the steps below:

1. Open the **Traffic Control** folder and double-click on the **Connection Setup** document. The Connection Administration dialog comes up.
2. Double-click on **<NEW>** to add a new connection. The Add a Connection dialog comes up.
3. In the Name field, type in SOCKS 1/2. The name is arbitrary, but the naming convention indicates that this is the first half of a SOCKS connection, originating from a source client and terminating at the firewall.
4. In the Description field, type in a descriptive name for the connection.
5. To identify the source of this connection, click on the **Select...** button associated with the Source field. This opens the Select Network Object dialog, which contains the objects you defined earlier. Select **Secure Network** and click on **OK** to close the dialog.

6. Now we identify the the termination point of this connection. Back in the Add a New Connection dialog, click on the **Select...** button associated with the Destination field. The Select Network Object dialog pops up again. This time, select **Secure Adapter** and click on **OK**.
7. Back in the Add a Connection dialog, move down to the Connection Services group and click on **Select...** We will now add a service associated with this connection. The Select a Service From the List dialog comes out.
8. Locate the predefined service SOCKS 1/2 on the list and highlight it. Click on **OK** to add this entry to the Connection Services list and return to the Add a Connection dialog.

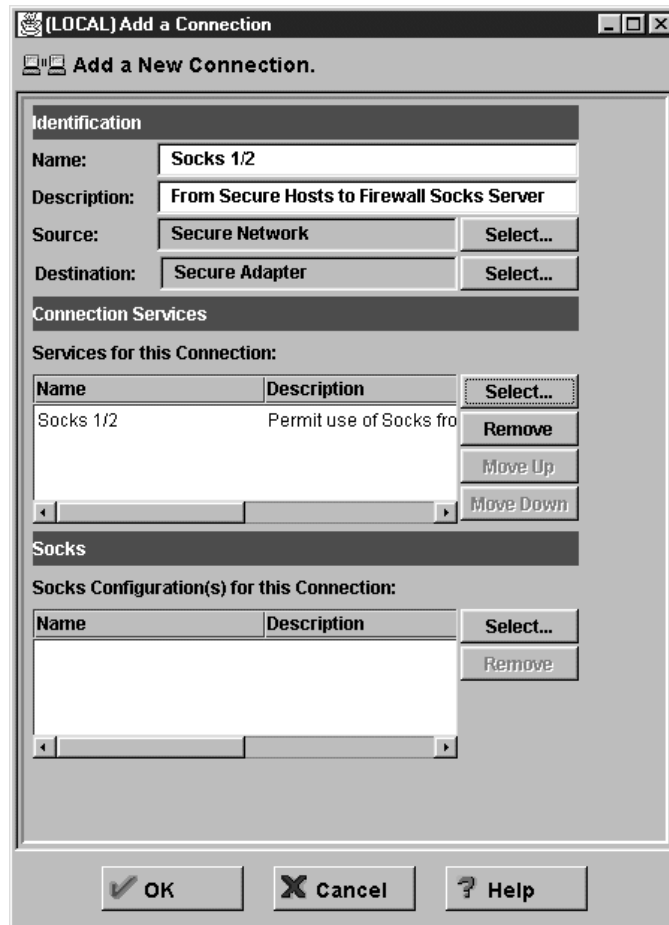


Figure 73. Creating the First Half of a SOCKS Connection

9. Click on **OK** to close the Add a Connection dialog. On the Connections Administration dialog, you will now see SOCKS 1/2 in the list.

We've defined half of the SOCKS connection. Now for the other half: from the SOCKS server on the firewall to the rest of the world:

1. From the Connection Administration dialog, add a new connection as before.
2. Name this new connection SOCKS 2/2 and enter its description. The Source for this connection will be the Non-secure Adapter and the Destination will be The World.

- In the Connection Services list, add the following predefined services: FTP proxy out 2/2, and HTTP proxy out 2/2. Confirm your changes and go back to the Connections List dialog.

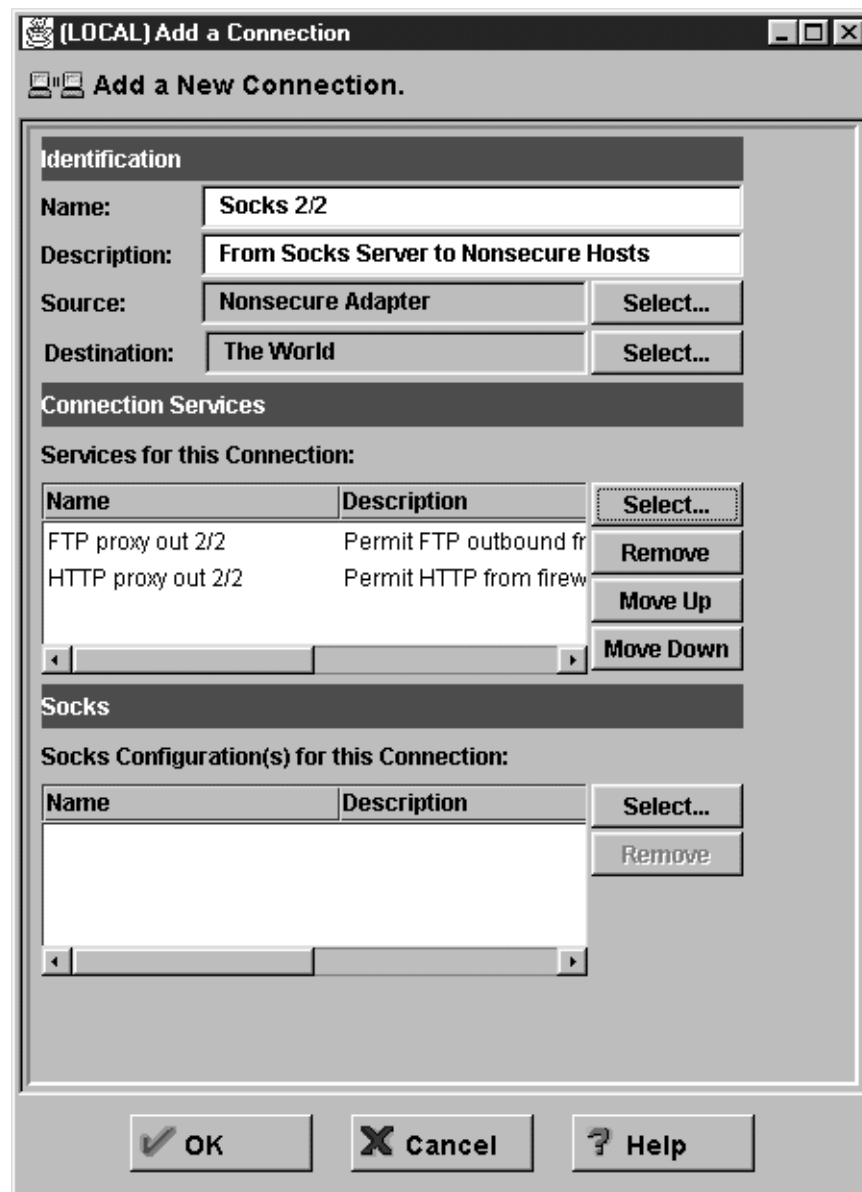


Figure 74. Creating the Second Half of the SOCKS Connection

Thus far, we have defined connections from the secure network to the SOCKS server and from the SOCKS server to the public Internet. But what about the SOCKS service itself; what connections should it permit? One more set of steps is necessary:

- In the Connection Administration dialog, create a new connection. Call this connection SOCKS; add a description as well.
- The connection source is the Secure Network and its destination is The World.
- Leave the Connection Services empty and move to the SOCKS subgroup. Click on the **Select...** button. This will bring up a dialog called Select a SOCKS Object From List....

- Highlight **FTP** and click on the **Apply** button. This will add the FTP SOCKS object to the connection list. Perform the same operation for the FTP PASV Data and HTTP selections.

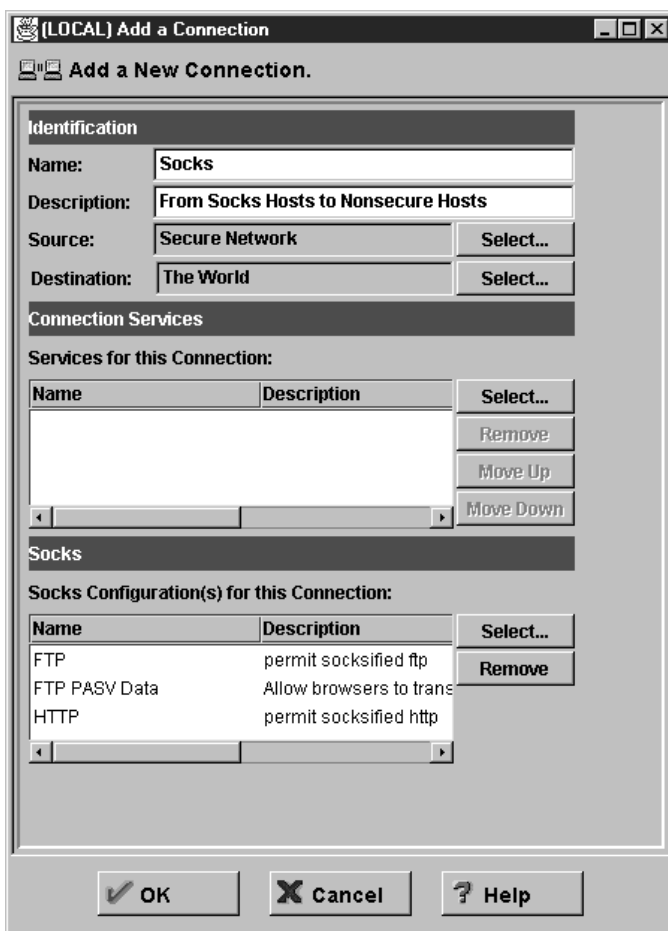


Figure 75. Applying SOCKS Rules

- Click on **OK** to confirm the choices and to close the dialog.
- Back on the Add a Connection dialog, click on **OK** to confirm the choice as well. This will bring you back to the Connections List dialog.

5.4.7 Reviewing and Activating the Connections

The SOCKS connections and services have been defined, but they are as yet not active. To enable the connections we have defined, you have to explicitly do so in the Connection Activation dialog which can be accessed from the Traffic Control folder. The Connection Activation dialog can also be accessed from other points of the Configuration Client, for example, the Connections List dialog box.

- Open the **Traffic Control** folder and double-click on the **Connection Activation** document.
- Select the **List Current Connection Rules** option and click on **Execute** to see the currently active rule sets. These rules, in tabular text format, comprise the configuration file of the packet filter of the IBM Firewall. To interpret these rules, please read Chapter 8, "IBM Firewall Rule Base" on page 143.

3. To activate the connection rules, select **Regenerate Connection Rules and Activate** button and click on **Execute**. The Output box gives you the results of the rules activation.

5.4.8 Configuring Alerts

As per 5.2, "Planning" on page 78 we are going to configure alerts for three events: changes made in the firewall's security policy, a malfunctioning SOCKS proxy, and occurrences of mail-related errors.

First, add an Alert Log facility to keep track of alert messages and display them on the Alerts Display window of the Configuration Client.

1. Open the **System Logs** subfolder in the System Administration folder.
2. Open the **Log Facilities** document. You will see in the resulting dialog box a list of active logging facilities. Currently, there is only one entry: Firewall Log, configured to record all events of info priority. This is the default entry.
3. Double-click on **<NEW>**. This will open the Add Log Facility dialog.
4. From the Facility drop-down list, select **Alert Log**. From the Priority drop-down list, select **Information**.
5. In the Log Filename field, type in C:\Program Files\IBM\Firewall\log>alert.log. All alert events will be written to this file. Close the Add Log Facility dialog and its parent, the Log Facilities dialog.

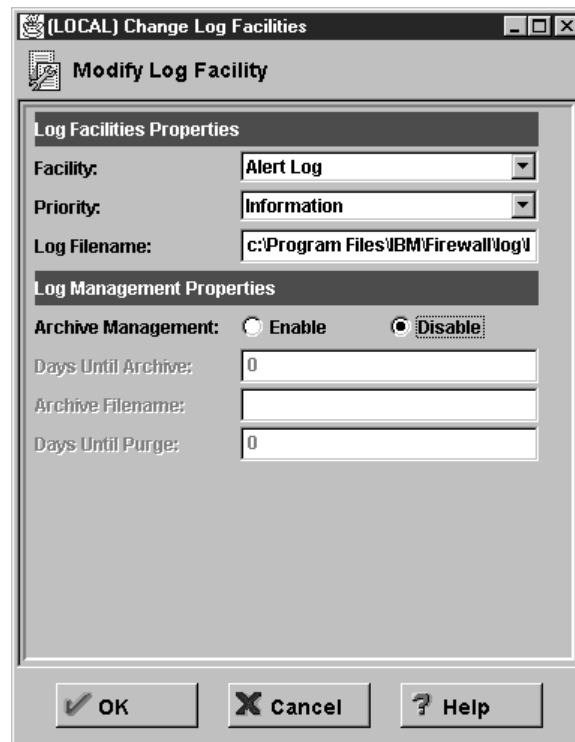


Figure 76. Creating the Alert Log

Administrators can now view messages from the Alerts Display of the main window of the Configuration Client. The buttons Latest and Previous below the Alerts Display message list scrolls up and down the list.

Next, configure log monitor thresholds. Thresholds will count how many times an event has occurred over a given period. If these limits are triggered, the firewall should perform some action such as notifying the administrator.

1. Back in the navigation tree, double-click on **Log Monitor Thresholds** to open the dialog. In the dialog's list box, you will see defined actions for the firewall. There will not be much to begin with.
2. Click on **<NEW>** to define a new threshold value. The Add Log Monitor will come up.
3. For Class Type, select **Message Threshold**. The Message Tag, Threshold Count, and Threshold Time fields become active.
4. The Message Tag defines the firewall events as identified by their unique IDs. The first message tag we want to enter is ICA1032, the ID for a Filter rules updated event. You can look up other message IDs in the *IBM eNetwork Firewall for Windows NT Reference Version 3 Release 3*.
5. We want to be notified of all ICA1032 filter rule update events, so in the Threshold Count field type in 1.
6. In the Threshold Time field, type in 1. Click on **OK** to close the dialog and add the new threshold.

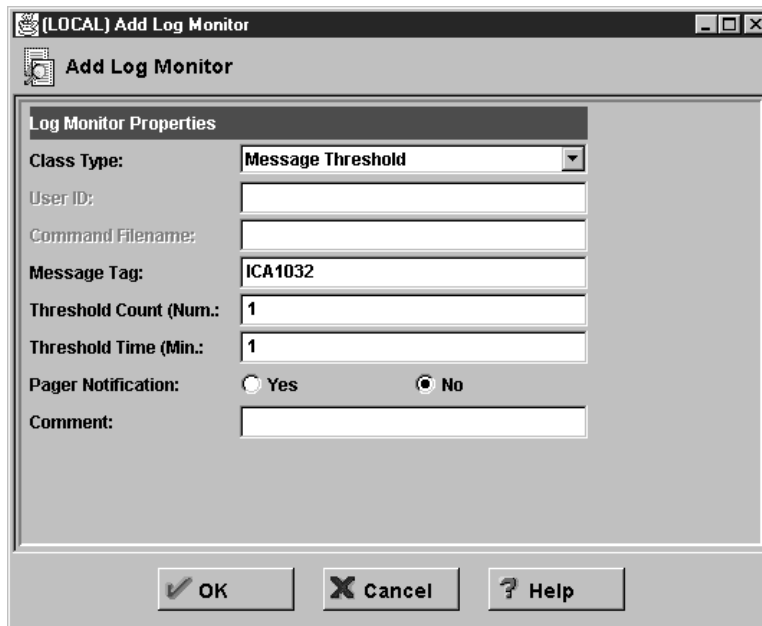


Figure 77. Setting a Message Threshold

7. Add the following set of message tags, which would indicate that the firewall is not accepting connections. These alerts will brace the administrators for the irate users who cannot surf the Web.

Table 11 (Page 1 of 2). Suggested Message Thresholds			
Message	Alert Description	Count	Time (Min)
ICA1034	Filter support deactivated	1	1
ICA1045	TCP overlimit	1	1
ICA1046	UDP overlimit	1	1

Table 11 (Page 2 of 2). Suggested Message Thresholds			
Message	Alert Description	Count	Time (Min)
ICA3012	SOCKS connection refused	1	1
ICA3127	SOCKS process terminated	1	1
ICA3130	SOCKS errors	1	1
ICA3135	SOCKS error-process terminate	1	1
ICA2164	SafeMail stop	1	1

- Now add the following message tags, which will alert the administrators that someone could be trying funny stuff with the SafeMail server.

Table 12. Suggested SafeMail Message Thresholds			
Message	Alert Description	Count	Time (Min)
ICA2180	SafeMail rejects sender	3	5
ICA2181	SafeMail rejects session	3	5
ICA2182	SafeMail rejects command	3	5

We want the firewall to alert the administrators by e-mail if there are any suspicious events. To do this:

- Click on **<NEW>** to define a new action. The Add Log Monitor dialog will come up.
- Create a new action of type Mail Notification.
- In the User ID field, type in the e-mail address of the administrator.

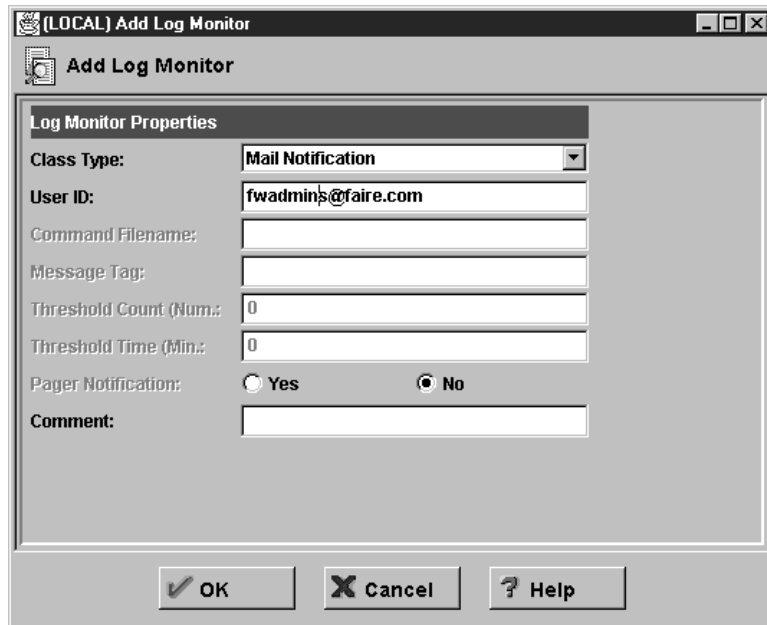


Figure 78. Creating a Mail Notification Object

Administrators' Mailing List

If you have more than one administrator, as is the case in this example, create a list address in the SMTP server that will forward mail to multiple administrators. Then, use this address in lieu of a single administrator's e-mail account in the User ID field.

5.4.9 Log Archiving

Firewall logs can grow very big very fast. As per the policy described in 5.2.2.6, "Archiving" on page 80, the logs are to be archived daily, and purged from the system every seven days. Currently, our system is maintaining two logs in the C:\Program Files\IBM\Firewall\logs directory: local4.log and alert.log. To manage these files:

1. Open the **Log Facilities Administration** dialog.
2. Double-click on the entry **Firewall Log**. In the Change Log Facilities dialog, look at the Log Management Properties. Enable the **Archive Management** option.
3. In the Days Until Archive, type in 1.
4. In the Archive Filename field, type in c:\Program Files\IBM\Firewall\log\local4.zip. In the Days Until Purge field, type in 7. Click on **OK** to confirm.

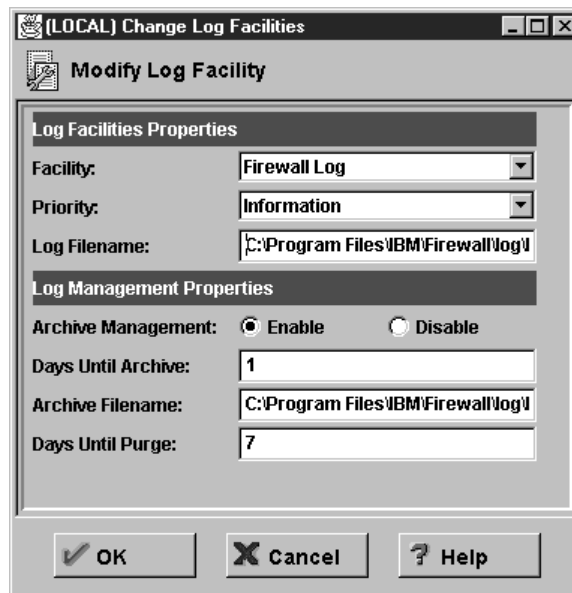


Figure 79. Archive Settings

5. Do the same for the Alert Log entry in the Log Facilities Administration dialog.

The task is not done yet. You must run `fwlogmgmt`, the IBM eNetwork Firewall for Windows NT log archiving program, on a regular basis corresponding to the settings you specified in the log facilities. The archiving program can be run automatically using Windows NT's scheduler service.

1. Open the **Services Control Panel** and locate the Schedule service. If you are working with a fresh installation of Windows NT and the IBM Firewall, the

Schedule startup should be at Manual. Change this to Automatic and start the service.

2. Open a command prompt window.

3. At the command line, type in:

```
at 3:00 /every:M,T,W,Th,F,S,Su fwlogmgmt -l
```

This archives the logs at 3:00AM every day.

4. At the command line, type in:

```
at 4:00 /every:Su fwlogmgmt -a
```

This purges the archived records every Sunday.

You can type `at` at the command line to see what jobs are in the Windows NT scheduler.

```
C:\>at
Status ID    Day                Time                Command Line
-----
          0    Each M T W Th F S Su  3:00 AM            fwlogmgmt -l
          1    Each Su              4:00 AM            fwlogmgmt -a

C:\>
```

Figure 80. Checking the Scheduled Jobs

You can delete a scheduled job by typing in `at <ID> /delete` where `<ID>` is the identification number assigned to the command. Typing `at /delete` without the ID deletes all scheduled jobs.

5.4.10 Configuring the Default Gateway

The default gateway for the firewall's non-secure adapter is the router at 204.189.54.102. You should already have configured this during the installation of the base Windows NT operating system.

5.5 Configuring Clients to Use the Firewall

Last of all, you must configure the clients in the secure network to use the firewall. The exercise is trivial; all you need to do is configure their DNS clients, default gateway settings, and their Web clients for SOCKS use.

5.5.1 Configuring DNS and the Default Gateway

The client workstations in Faire.Com must be configured to use the DNS and default gateway in the same way that the servers are. Refer to 5.3, "Configuring Peripheral Services" on page 81 for details.

5.5.2 Configuring Web Clients

Web clients such as Netscape Navigator and Internet Explorer must be configured to use SOCKS. The SOCKS host, in this case, will be blue.faire.com and the SOCKS port will be 1080. (If you want a more descriptive name, create a CNAME alias for blue.faire.com in the secure name server. A good example would be socks.faire.com).

Using Netscape Navigator 4.0 as an example:

1. Open **Netscape Navigator**. From the Edit pull-down menu, click on **Preferences**. This will open the Preferences dialog.
2. Open the **Advanced** branch in the Category tree of the Preferences dialog. Click on **Proxies**.
3. On the right panel, select the **Manual proxy configuration** option and click on **View....** This opens the Manual Proxy Configuration dialog.
4. In the SOCKS field of the Servers group, type in `blue.faire.com`. Leave the port number 1080 as is.

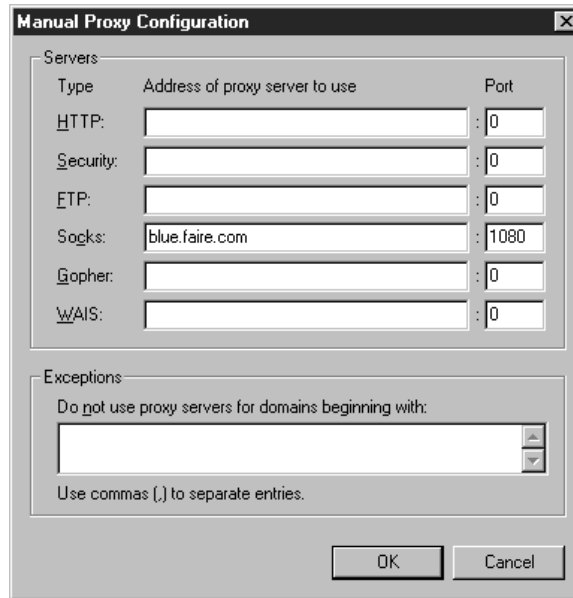


Figure 81. Configuring Web Clients for SOCKS

5.6 From Here On...

The objective of this chapter has been to show you how to configure the firewall for the typical Internet security needs of a small company. We started with a description of the needs of our case study, Faire.Com, and from that formulated a very basic security policy. From the security policy, we proceeded to configure the IBM Firewall for the required services. We also configured other systems with direct bearing on the operation of the firewall.

In real life, companies may require configurations somewhat more complex than what we have presented here. For this reason, you should read the following chapters.

- For an in-depth discussion of IP filters as they are used in the IBM Firewall, read Chapter 8, “IBM Firewall Rule Base” on page 143 and Chapter 9, “Examples of Rules for Specific Services” on page 161.
- For an in-depth discussion of proxies and SOCKS, refer to Chapter 13, “Configuring Proxy Services and SOCKS” on page 293.

- For more options with DNS, read Chapter 11, “Domain Name Service (DNS)” on page 247.
- For more options with mail, read Chapter 12, “Mail Handling” on page 269.

Chapter 6. Firewall Setup Using the Wizard

The setup wizard aids you with the initial configuration of the firewall. It appears automatically after you log onto the firewall for the first time. Thereafter, the setup wizard is available under the Help menu item on the GUI. The usage of the wizard is optional; you are not required to use it to configure the firewall. Use it if you want to get your firewall up and running quickly.

The wizard is mainly intended for novice Firewall users; experienced users may want to use the GUI to configure the firewall.

As any wizard, this one should not require a major explanation; it should be intuitive. We are including this chapter so that you can quickly browse through the wizard functions and quickly decide on whether to use it.

6.1 Wizard Objectives

The setup wizard guides you through the following fundamental tasks:

- System administration tasks having to do with security policies, interfaces, DNS, mail and log setup
- Setup to allow secure users to access the nonsecure network through the Web, Telnet and/or FTP
- Setting authentication methods for the default user based upon services that they enabled with the wizard
- Creating an alert log
- Setting up some basic log monitor thresholds
- Activating filter rules

The setup wizard can be helpful for getting started on a variety of firewall installations. However, depending upon your circumstances, the wizard may not be recommended for:

- Multiple network adapters on the secure side of the firewall for which there is more than one security policy
- Migrating a configuration from a previous version of the firewall

6.2 Wizard Sample Configuration

The following screens guide you through the process of a quick basic firewall configuration:

For the test we are using our test scenario as shown in Figure 270 on page 285.

Secure Interface:	9.24.104.57
Nonsecure Interface :	150.53.104.56
Internal Domain Name:	itso.fair.com
Internal DNS Server Name:	dns.itso.fair.com
External Domain Name:	fair.com
Secure Mail Server Name:	mail.itso.fair.com

Figure 82. Naming Convention from Our Test Scenario

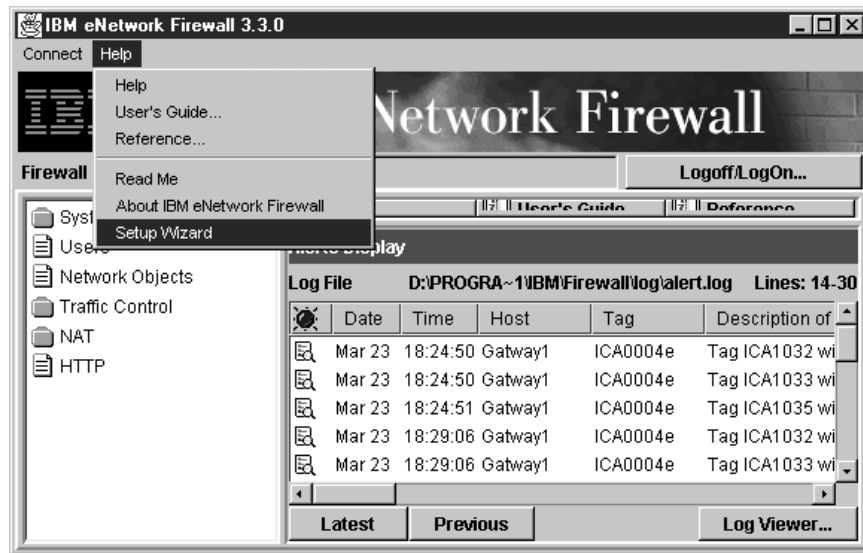


Figure 83. Firewall Main Panel

Step 1: Wizard start.

From the Main panel select **Help** and then **Setup Wizard**.

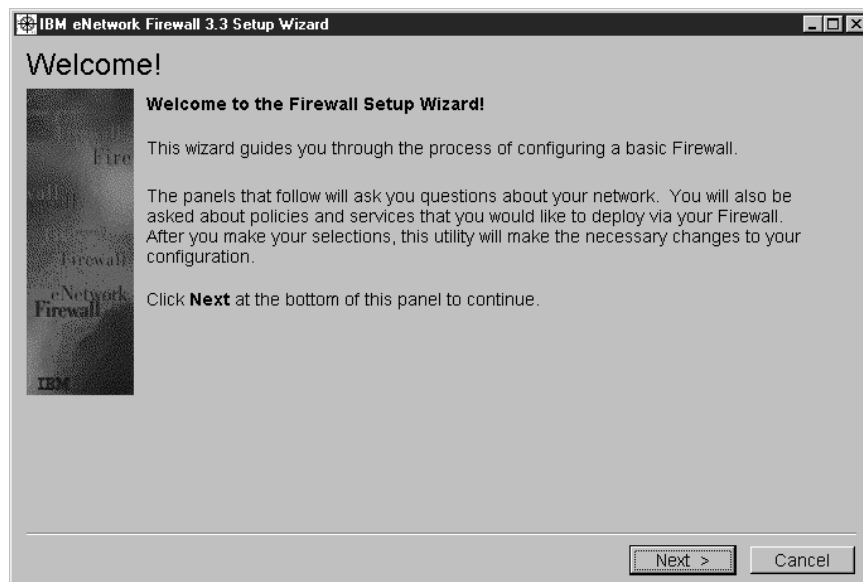


Figure 84. Welcome Screen

Click on **Next**.



Figure 85. What to Expect Screen

Follow the screens step by step.

Click on **Next**.



Figure 86. Important Notice

The wizard clearly states in which cases its use is not recommended.

Click on **Next**.



Figure 87. Warning Screen

Step 2: Warning !

The wizard is primarily intended for initial configuration. However, if you have already defined connections, you will see this warning screen. From now on previous wizard connections will be overwritten or deleted in case you decide to cancel the wizard at a later step.

Click on **Next**.

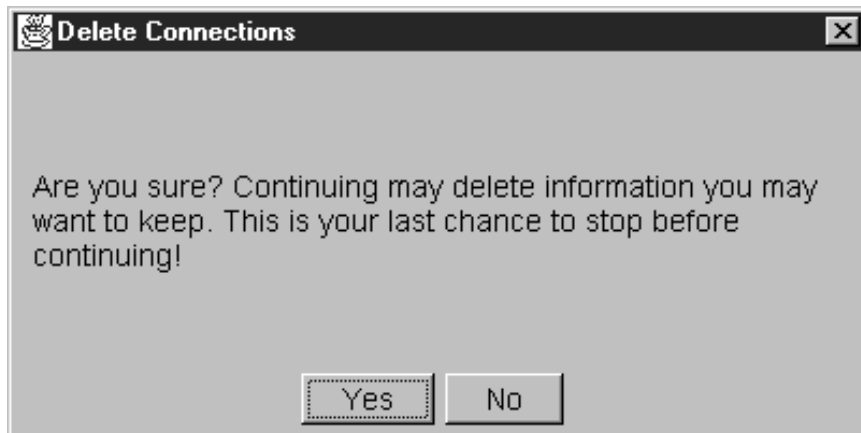


Figure 88. Delete Connection Screen

Final Warning!

Click on **Yes** when you have made up your mind.

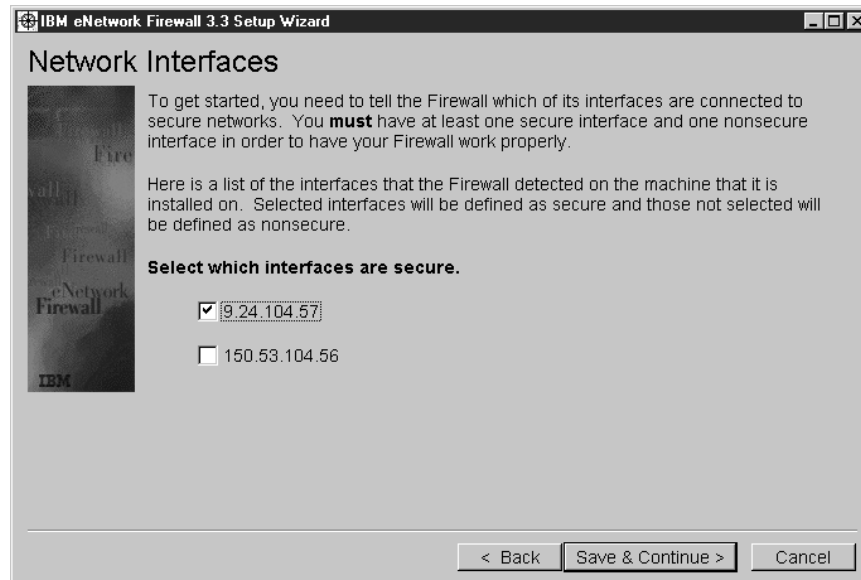


Figure 89. Network Interface Configuration

Step 3: Interface Selection.

Here you separate your company network from the Internet. Mark your secure interface. If your system discovers more than two interfaces and you indicate two interfaces as secure, in the next screen you get the warning that the wizard will set up the same policy for both interfaces.

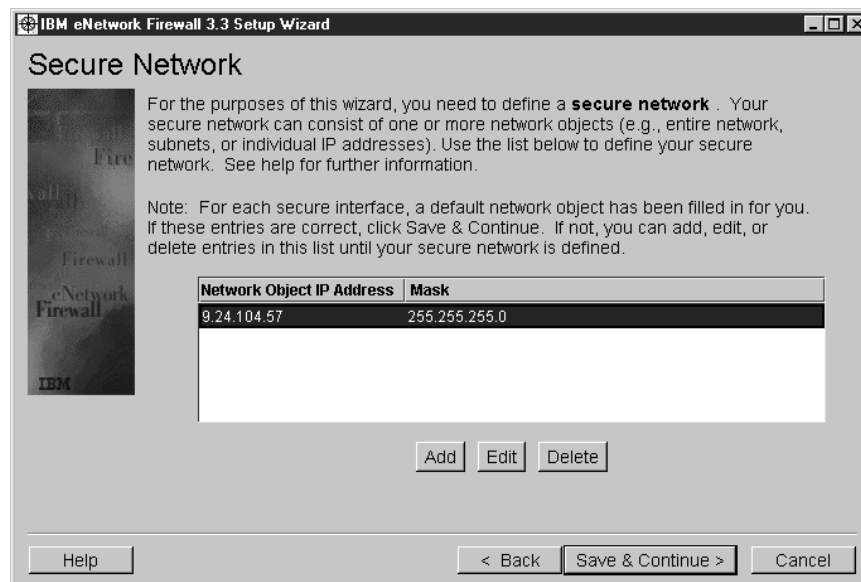


Figure 90. Secure Network Configuration

Step 4: Define Secure Network(s).

The objective of this screen is for you to tell the wizard which is your secure network. Note that the wizard has pre-filled a network object definition based upon your secure interface. The idea here is that the wizard makes *best guesses* as to which is your secure network. For example, in the figure above it is guessing that your secure network is any IP address that starts with 9.24.104. If this subnet

sufficiently describes your secure network, click on **Save & Continue**. If not, click on **Add**, **Edit** or **Delete** to define your secure network.

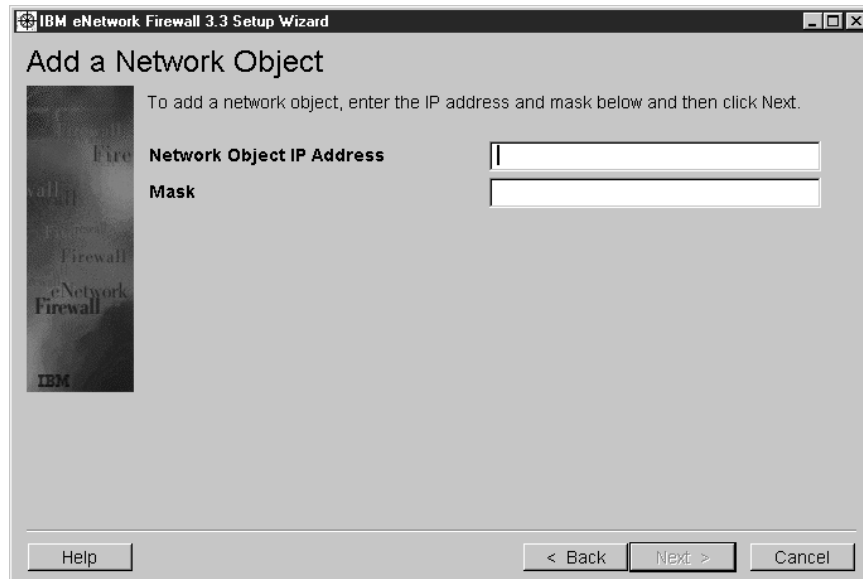


Figure 91. Add Secure Network(s)

Click on **Next** if you have added your secure network.



Figure 92. Secure Domain Name Service

Step 5: Domain Name Service Configuration.

Enter the name of the domain that is protected by the firewall. This is the name of your secure domain network.

Click on **Next**.

Note that you may click on **Skip Section**. This button allows you to defer configuring your DNS settings for now.

Figure 93 on page 115 shows what happens if you had clicked on **Skip Section** in Figure 92.

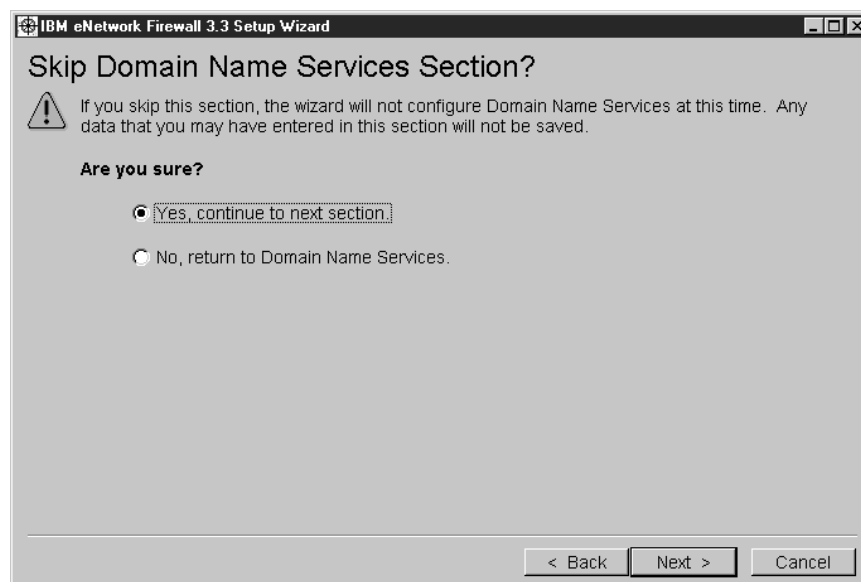


Figure 93. Skip Screen

The wizard continues with the DNS configuration.



Figure 94. Secure Domain IP Address

Step 6: DNS Configuration Secure IP Address.

If you are using the wizard for the first time click on **Add** and enter the IP address of the secure DNS server. You may have several. Otherwise click on **Next**.

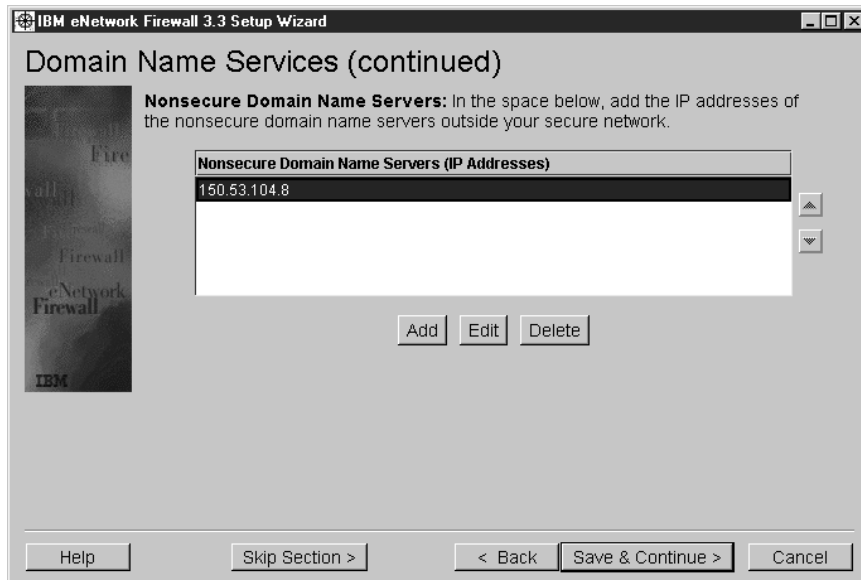


Figure 95. Nonsecure Domain IP Address

Step 7: Secure Mail Proxy Configuration.

Again, if you are using the wizard for the first time click on **Add** and enter the IP address of the nonsecure DNS server. Otherwise click on **Save & Continue**. The wizard will now do the Secure Mail Proxy configuration.



Figure 96. Secure Mail Proxy

Step 8: Click on **Add**.



Figure 97. Add Secure Mail Proxy

Step 9: Add a Mail Server.

Enter your secure domain name, mail server name, and public domain name. Click on **Save & Continue**.



Figure 98. Display Secure Mail Proxy

Step 10. Maintaining Mail Server Configuration.

You may enter additional mail servers, change or delete. Click on **Add**, **Edit** or **Delete** according to what you want to do.

Click on **Next**.

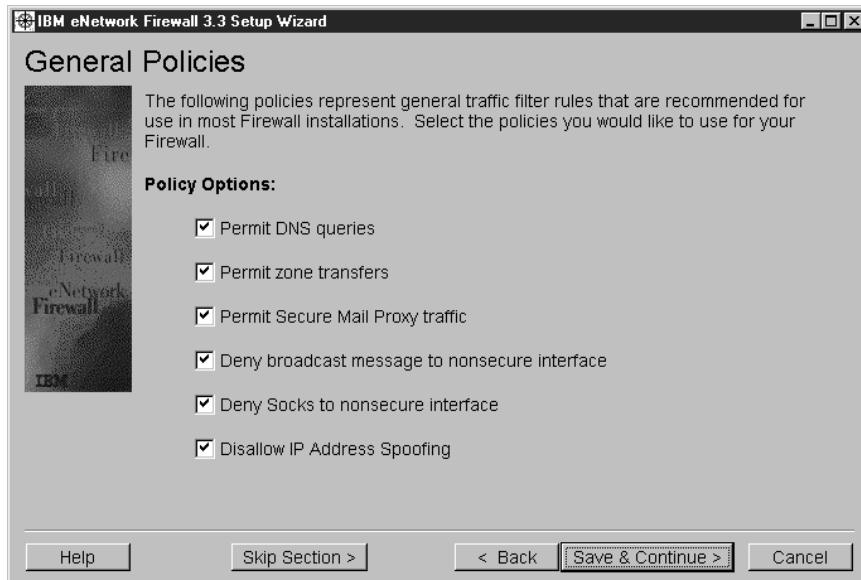


Figure 99. Security Policies Configuration

Step 11: General Policies.

The marked options that you see under Policy Options are recommended for most firewall installations. If you are not sure what they mean, use the Help button or read Chapter 9, “Examples of Rules for Specific Services” on page 161.

Click on **Save & Continue**.

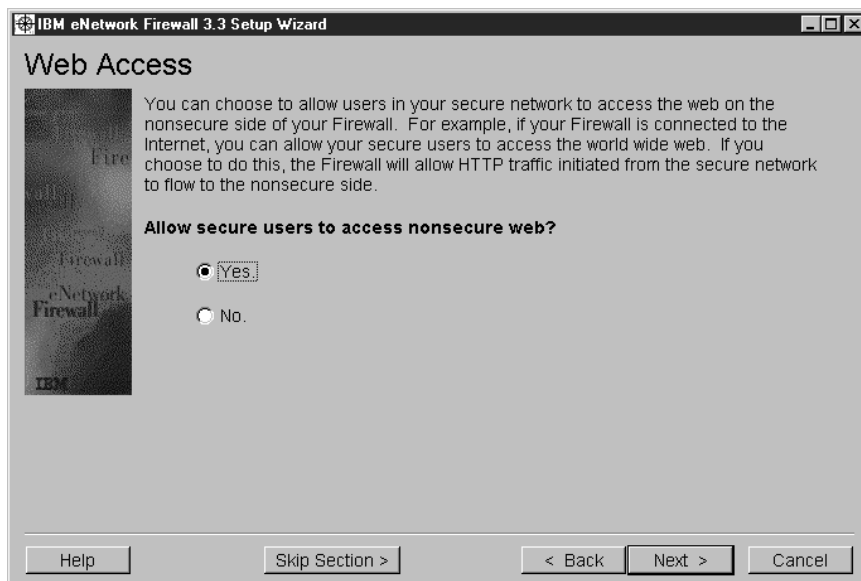


Figure 100. Secure User Access to the Web

Step 12: Web Access.

If you allow all your secure users to access the Internet, click on **Yes**. If you want to give Web access to some of the users, but not all, you cannot use the wizard for this purpose.

Click on **Next**.

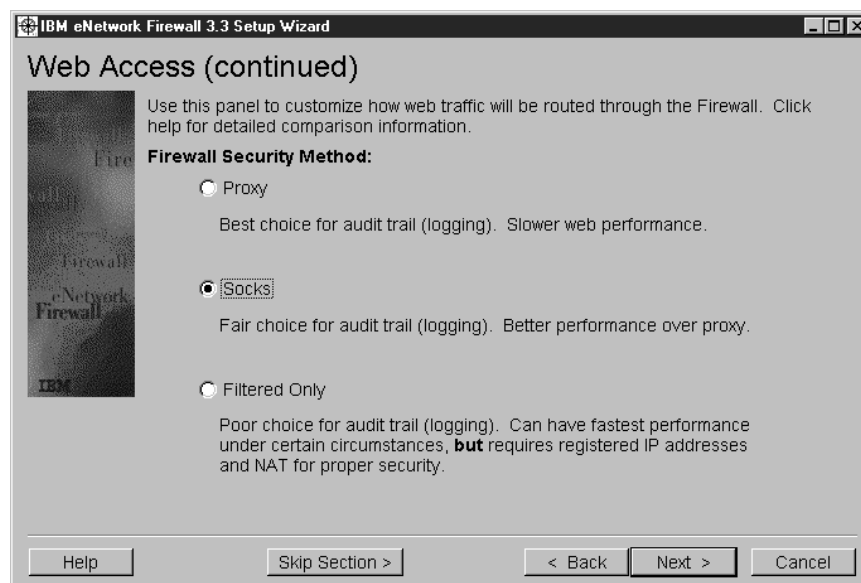


Figure 101. Web Access via Proxy, Socks or Filter

Step 13: Customize Web Access.

Customize how Web traffic will be routed through the firewall; either via HTTP proxy or Socks or routed traffic. If you are in doubt leave Socks as the default otherwise refer to Chapter 13, “Configuring Proxy Services and SOCKS” on page 293 to get further information. If you want to use a mix of these methods (for example, HTTP proxy and Socks) you cannot do it with the wizard.

Click on **Next**.

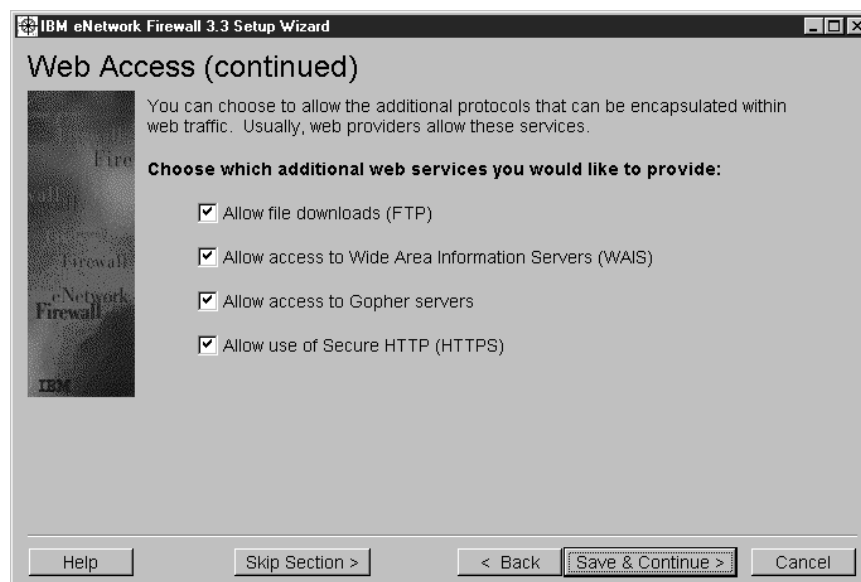


Figure 102. Selection of Additional Services

Step 14: Configuring Services.

Select the appropriate services you would like to provide. These are the services that are running basically under cover within HTTP, FTP for file downloading, for example, or various search services. Usually providers allow these services. See Chapter 9, “Examples of Rules for Specific Services” on page 161 for more information.

Click on **Save & Continue**.



Figure 103. Telnet Access

Step 15: Telnet Access.

Do you want to allow secure users to Telnet to the nonsecure side? Choose accordingly.

Click on **Next** with No as the default.

If you click on **Yes**, leave proxy as the given default as seen in the next screen; otherwise refer to Chapter 13, “Configuring Proxy Services and SOCKS” on page 293 for further information.

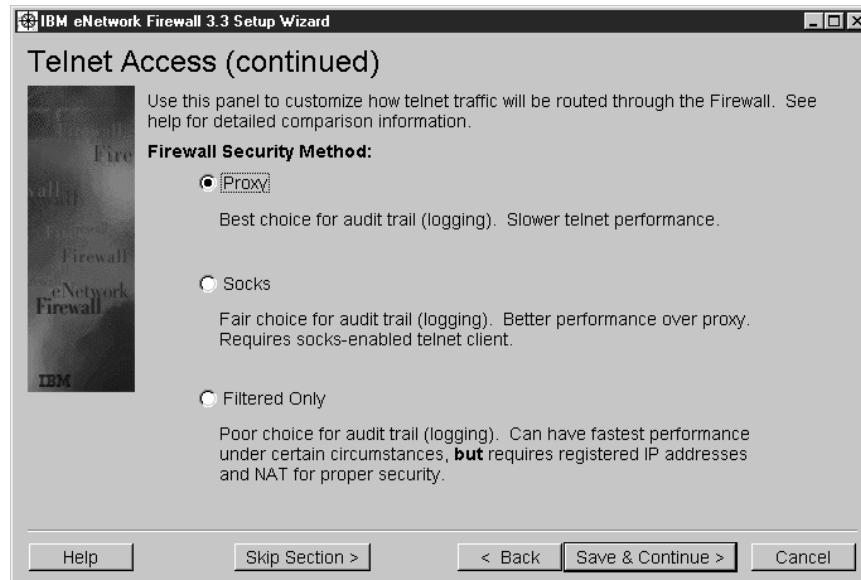


Figure 104. Telnet Access Service Selection Screen

Click on **Save & Continue**.

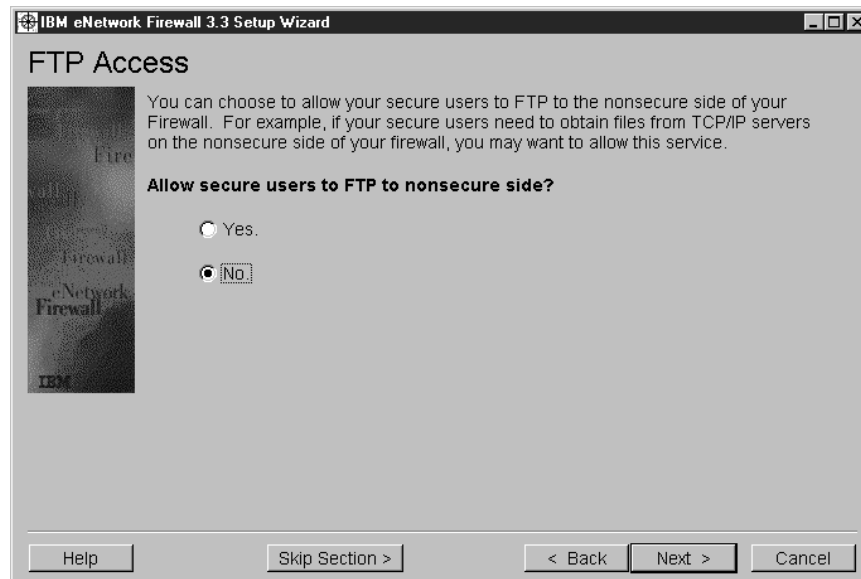


Figure 105. FTP Access

Step 16: FTP Access.

Do you want to allow all secure users to FTP to the nonsecure side? Choose accordingly.

Click on **Next** with No as the default.

In case you click on **Yes** leave Proxy as the given default as seen in the next figure, otherwise refer to Chapter 13, “Configuring Proxy Services and SOCKS” on page 293 to get further information.

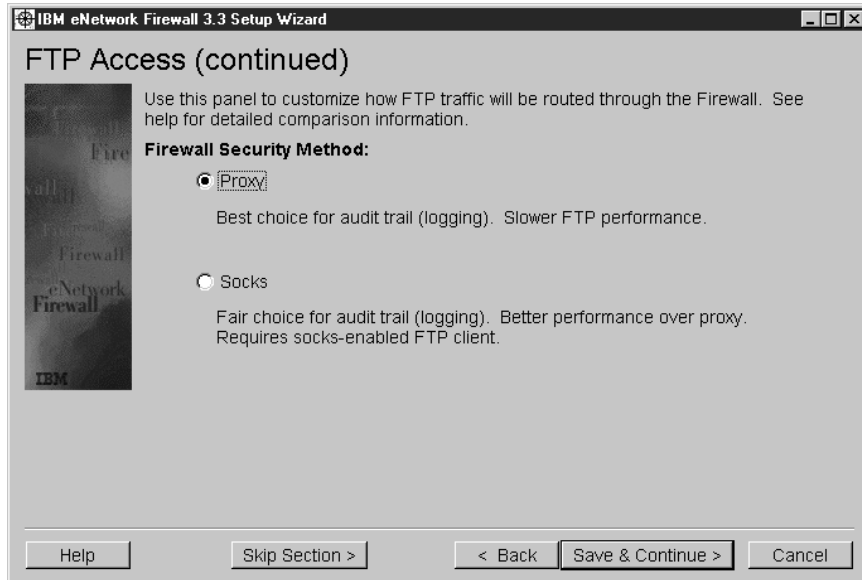


Figure 106. Telnet Access Service Selection Screen

Click on **Save & Continue**.



Figure 107. Firewall Log File Creation

Step 17: Firewall Log.

Messages generated by the firewall are logged in the appropriate file. The IBM Firewall uses five priority levels:

- debug
- information
- warning
- error
- critical

For a detailed discussion on priority levels, please refer to Chapter 16, “Logging, Monitoring, and Reporting” on page 409.

Click on **Save & Continue**.



Figure 108. Alert Log

Step 18: Alert Log.

The alert log collects warnings about abnormal activities on the firewall. A log file will be created as shown under Log Filename. The priority levels from the firewall log (previous step) and for the alert log are identical. For a detailed discussion on these levels please refer to Chapter 16, “Logging, Monitoring, and Reporting” on page 409.

Click on **Save & Continue**.

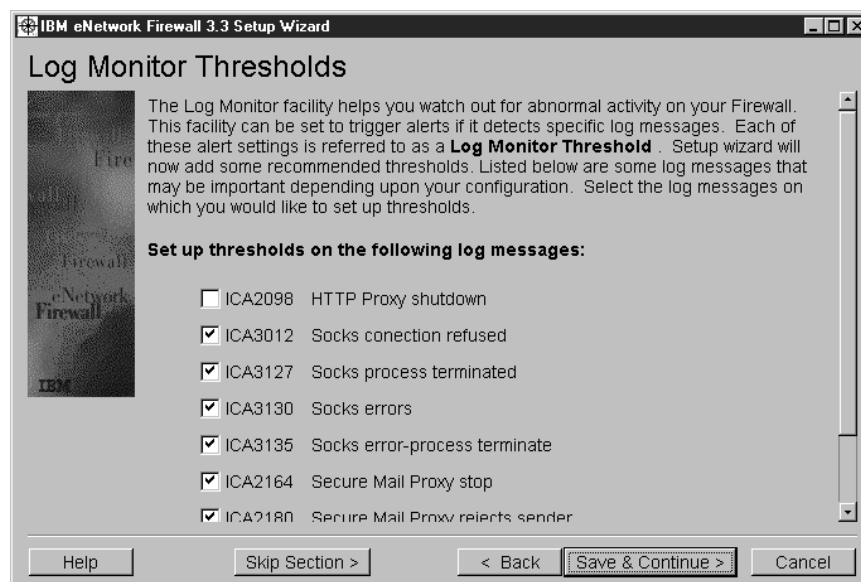


Figure 109. Log Monitor Thresholds

Step 19: Log Monitor Thresholds.

The triggered alerts are referred to as a log monitor threshold. These messages should be logged according to your security policy. Leave the defaults or choose accordingly.

A threshold consists of a count and a time parameter. A threshold is violated if a number of specific events (count) is exceeded in the specified amount of time (time). For a detailed discussion on Log Monitor Thresholds please refer to 16.2.1, "Log Monitor Thresholds" on page 420.

Click on **Save & Continue.**

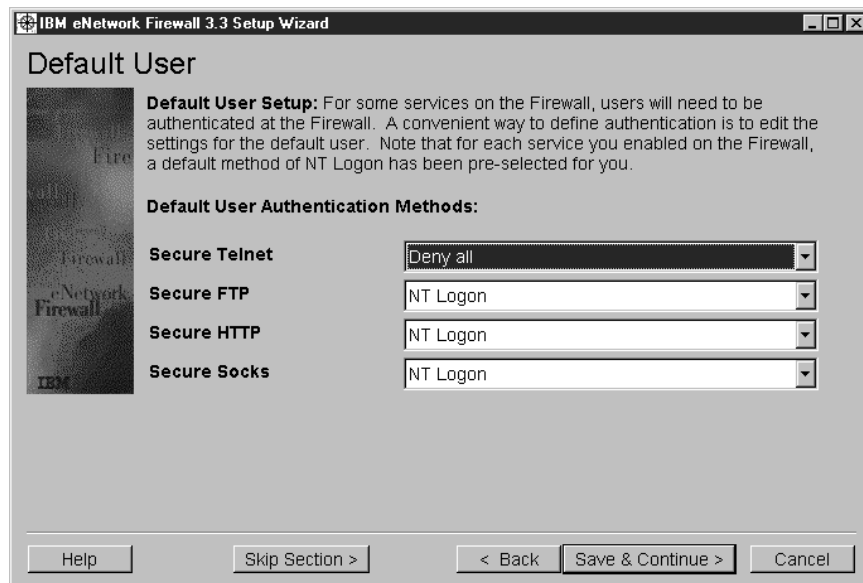


Figure 110. Default User Setup

Step 20: Default User.

For some services a firewall user will need to be authenticated.

The firewall comes with a pre-installed user called *fwdfuser*. The default user setup of the wizard relates to this *fwdfuser* and is used to determine the user's firewall attributes, such as which authentication methods to use when authenticating the user. For example, if a user is not defined in the firewall user database but is an NT user, and tries to use the firewall proxy services, he or she will be subjected to the attributes and authentication methods of the default user.

Click on **Save & Continue.**



Figure 111. Activation

You can choose to activate your configuration now or at a later time. If you want to test what the wizard has done, you may want to activate now. Or you may want to use the wizard's configuration as a starting point and continue configuring with the GUI.

Click on **Finish**.

Chapter 7. Coming to Grips with IP Packets

As we have described, a firewall needs to intercept every IP packet that it receives and then process it according to the policy that you define. In this chapter we discuss the packet characteristics that the firewall can use to make processing decisions.

The IBM Firewall machine operates between two or more IP networks, sometimes acting as a router (passing packets between the secure and non-secure sides), sometimes providing a proxy server function to break the session in two. The IBM Firewall uses IP filters to control which packets are passed and which are blocked on each side.

The information it uses to decide whether to block or pass a packet is largely contained in the packet headers. Some of the filtering criteria are:

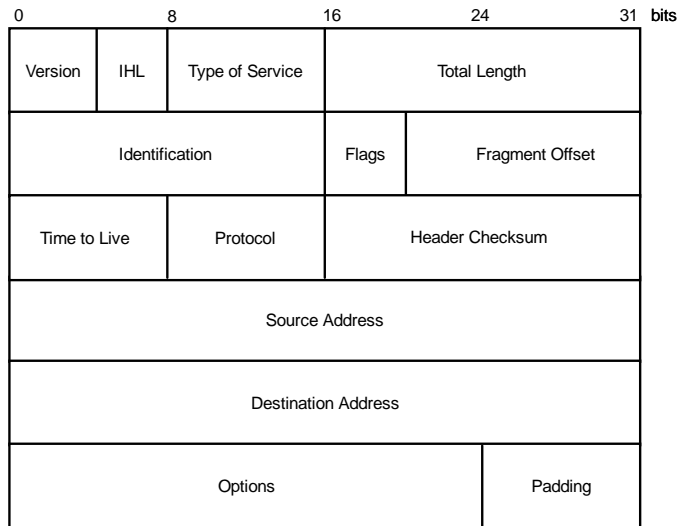
- The source and destination IP address
- The direction of flow
- The IP protocol (ICMP, TCP, UDP or other protocols)
- The interface where the packet is detected (secure or non-secure)

Before we show detailed examples of how to set up the firewall controls, we will consider the characteristics of the different IP protocols to gain a better understanding of what the firewall looks for.

7.1 An Introduction to IP Packets

Normally, a firewall does not look at the content of a packet, but only at the header. An IP packet consists of a formatted header, followed by the packet payload. The payload itself may include a further header containing session-level protocol information (for example, a TCP or UDP header).

Figure 112 on page 128 shows the format of the IP packet header.



5209\PIPP001

Figure 112. IP Packet Header Format

The following are the important fields in the IP header:

- The source address
- The destination address
- The fragmentation indicator (in the flag field)
- The protocol ID

The source address, destination address and protocol ID are used by the firewall filters to define which machines can access which particular service.

The fragmentation indicator is used to instruct IBM Firewall on how to handle fragmented packets. Different types of networks support different maximum packet sizes, so sometimes a router has to break a packet into smaller fragments to pass it from one network to another. The packet-filtering firewalls have to be aware of packet fragmentation because only the first fragment contains the header information of higher-layer protocols, such as UDP and TCP. Later fragments in a packet could override header fields, such as the source and destination port.

The IP specifications allow packets of very small sizes. The minimum packet size that can be sent according to RFC 791 is 68 octets. The problem here is that this packet size is not enough to carry the complete information for upper layer protocols. This leads to an attack technique called the *tiny fragment attack*.

The reassembly algorithm contains a mechanism by which later fragments can overwrite the data portions of previous fragments. An attacker could create a series of packets in which the first fragment will be allowed by the filter, but later fragments will overwrite relevant information (such as TCP source and destination ports). In this way the filtering rules can be bypassed if you allow fragmented packets. This is called the *overlapping fragment attack*.

Ideally you should configure your firewall to only support nonfragmented packets. See *RFC 1858 Security Considerations for IP Fragment Filtering* for a complete discussion about this point. Also both attacks are described in that RFC.

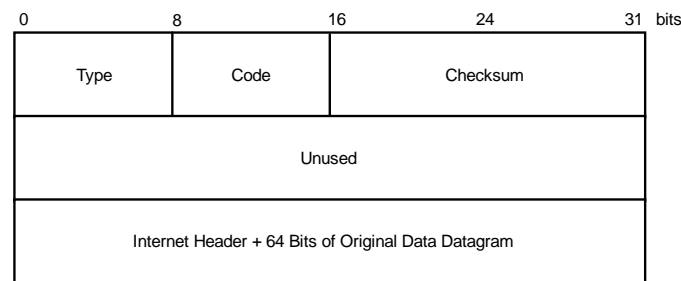
7.2 An Introduction to ICMP Packets

ICMP is a protocol designed to communicate errors and information between hosts that are processing IP datagrams. You can find the specification of ICMP in RFC 792. It is used for purposes such as informing that a host is unreachable or that a sender is sending packets too fast.

The ICMP messages that most people are familiar with are the ones that are generated by the ping command, but in fact there are many different types of ICMP messages. Ping generates an ICMP *echo request* message and expects to receive an *echo reply* message in response. Echo request is a relatively safe message, but any of the ICMP messages can be used by an outsider in order to gain some knowledge of your network or to directly attack your system. Also, like every protocol that you allow, ICMP messages can be used to overwhelm your systems in a *denial of service* attack.

Each ICMP message consists of a type plus a code, both of which are small integer values. Unlike the higher layer protocols, such as TCP or UDP, there is not a source port nor a destination port, just the message type and code.

Every ICMP message has the following format:



5209\PIPP002

Figure 113. ICMP Packet Format

When configuring firewall filters you *could* disable all ICMP messages in both directions if you don't care about the different types of message. This may make it difficult for you and your users to troubleshoot access problems, but will be safer and simpler for you. You also have to consider that some ICMP messages are used by network management applications (principally echo and address mask).

We will now look at each of the ICMP message types. For each message type we describe ways in which it could be abused by an attacker and suggest a suitable filtering policy. We show examples of the firewall connection definitions to implement our suggestions in 9.1, "What Services Should You Provide?" on page 161. In Appendix F, "ICMP Message Types and Codes" on page 615 is a summary of all the ICMP message types and codes, including RFC information where appropriate.

7.2.1 Echo and Echo Reply Messages

Type	Description	Code
8	Echo	0 - No code
0	Echo reply	0 - No code

Description: The echo (also called echo request) message is used to check if a host is up or down. When a host receives the request, it sends back an echo reply message. These messages are usually generated by a ping command, but may also be generated by a network management station that is polling the nodes of a network.

Firewall Considerations: Echo request can be used by an outsider to map your network. We suggest you allow the outgoing echo request and incoming echo reply. Disable the incoming echo request and outgoing echo reply.

You could consider enabling this facility to some key hosts, such as the router of your network provider. You might allow incoming pings to the non-secure adapter of the IBM Firewall.

7.2.2 Destination Unreachable Message

Type	Description	Code
3	Destination unreachable	0 - Net Unreachable 1 - Host Unreachable 2 - Protocol Unreachable 3 - Port Unreachable 4 - Fragmentation Needed and DF Set 5 - Source Route Failed 6 - Destination Network Unknown 7 - Destination Host Unknown 8 - Source Host Isolated 9 - Communication with Destination Network is Administratively Prohibited 10 - Communication with Destination Host is Administratively Prohibited 11 - Destination Network Unreachable for Type of Service 12 - Destination Host Unreachable for Type of Service 13 - Communication Administratively Prohibited by Filtering 14 - Host Precedence Violation 15 - Precedence Cutoff in Effect

Description: These messages are generated by hosts or intermediate routers in order to notify that a session cannot be established.

Firewall Considerations: An outsider can force nodes of your network to generate these packets in order to obtain knowledge of your network. For example, they can use a port scanner to learn which services you are providing. If you reply with a port unreachable, they will know that you are not providing this service. This type of information can also be gathered for TCP services by using stealth scanning (see 7.4.2, "Stealth Scanning" on page 141).

You should receive these messages, as they may provide useful information for troubleshooting. You should only send them through the secure interface, because if you send them through the non-secure interface, it will help outsiders to map the services that you are offering.

7.2.3 Source Quench Message

Type	Description	Code
4	Source Quench	0 - No code

Description: This message is generated by a host or a router when it wants the sender to slow down the rate at which it is sending packets. The IP stack passes this packet to the upper layers, and they are responsible for slowing the rate down.

Firewall Considerations: This message could be used by an attacker (probably combined with IP spoofing) in order to make a very effective denial of service attack. Unfortunately it is more often a legitimate message, so if you decide to filter it, you may cause problems due to lost packets. We suggest you allow it to be sent and received, but also log the received messages for later analysis.

7.2.4 Redirect Message

Type	Description	Code
5	Redirect	0 - Redirect Datagrams for the Network 1 - Redirect Datagrams for the Host 2 - Redirect Datagrams for the Type of Service and Network 3 - Redirect Datagrams for the Type of Service and Host

Description: This message is generated by a router when it receives a packet from a host and forwards the packet to another router that is on the same network as the host from which it received the packet (not the original sender, the last hop sender). This message is intended to modify the routing table of the receiver so that the router does not have to do unnecessary work. The example in the following figures helps to explain how this works:

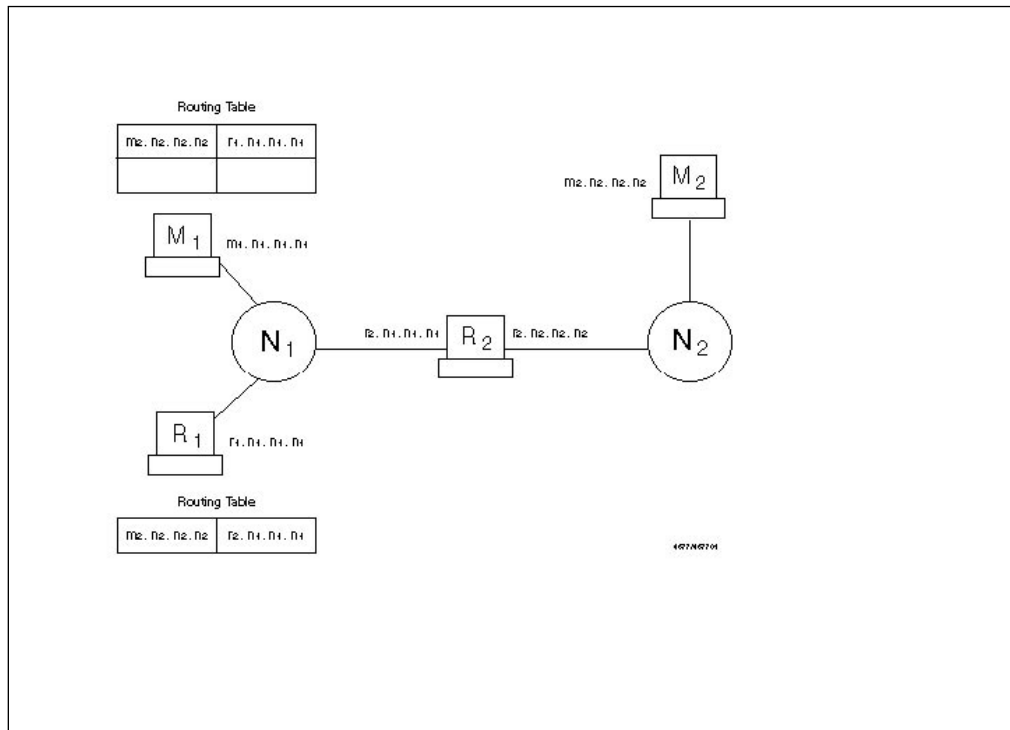


Figure 114. ICMP Redirect, Configuration

Suppose that machine M1 in network N1 wants to send a packet to machine M2 in network N2. As it is not directly connected to network N2, it looks at its own routing table in order to find who it must send the packet to. So it sends the packet to router R1 (see Figure 115).

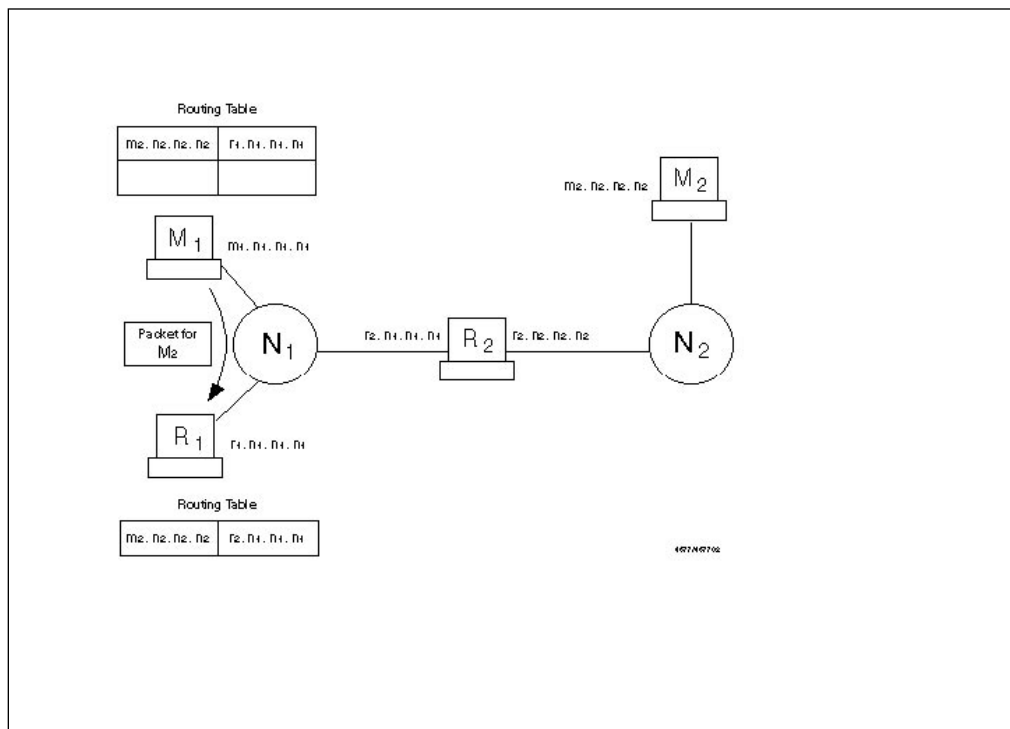


Figure 115. ICMP Redirect, M1 Sends Packet to R1

Router R1 receives the packet. As it is not directly connected to network M2, it looks at its own routing table in order to find who it must send the packet to. It forwards the packet to router R2 (see Figure 116 on page 133).

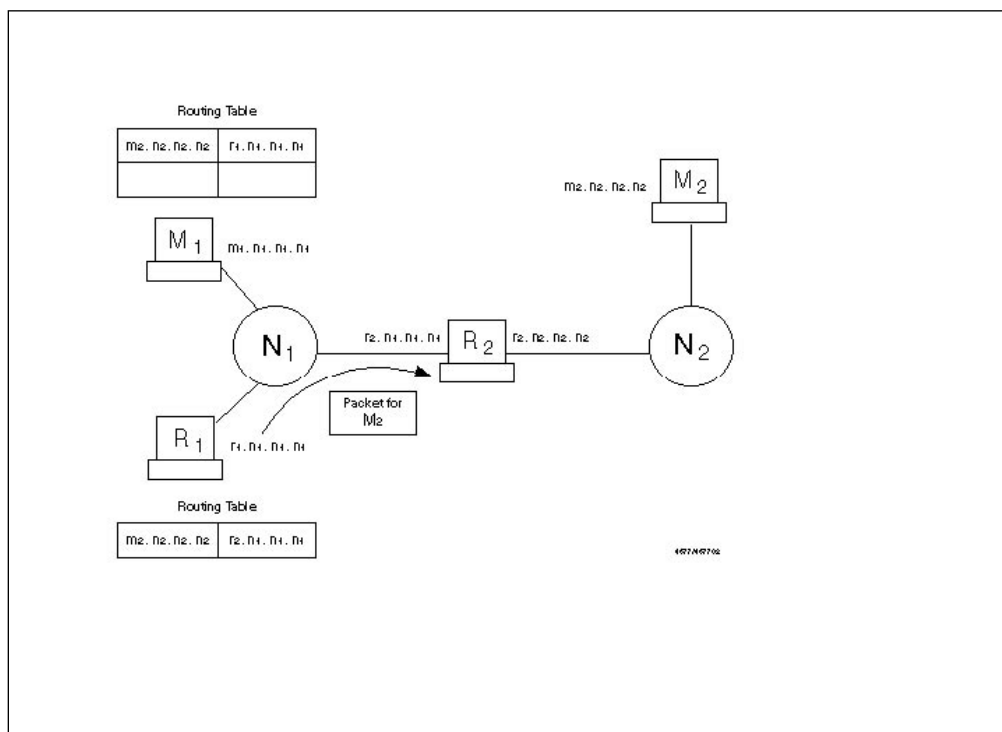


Figure 116. ICMP Redirect, R1 Sends Packet to R2

Router R1 realizes that it sent the packet through the same interface on which it was received. So instead of R1 receiving messages for M2 from M1 and then resending them to router R2, it sends a redirect message to M1 telling it to use R2 as the router in order to reach M2 (see Figure 117 on page 134).

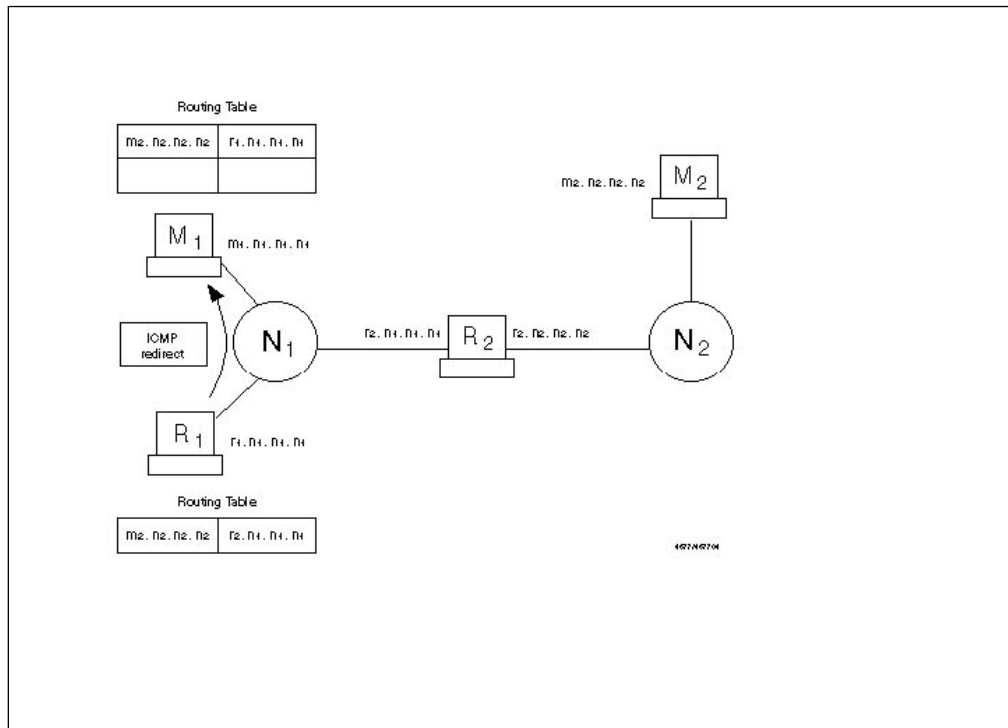


Figure 117. ICMP Redirect, R1 Sends Redirect Message to M1

Machine M1 receives the ICMP redirect message from R1 and updates its routing table in order to be more efficient (see Figure 118).

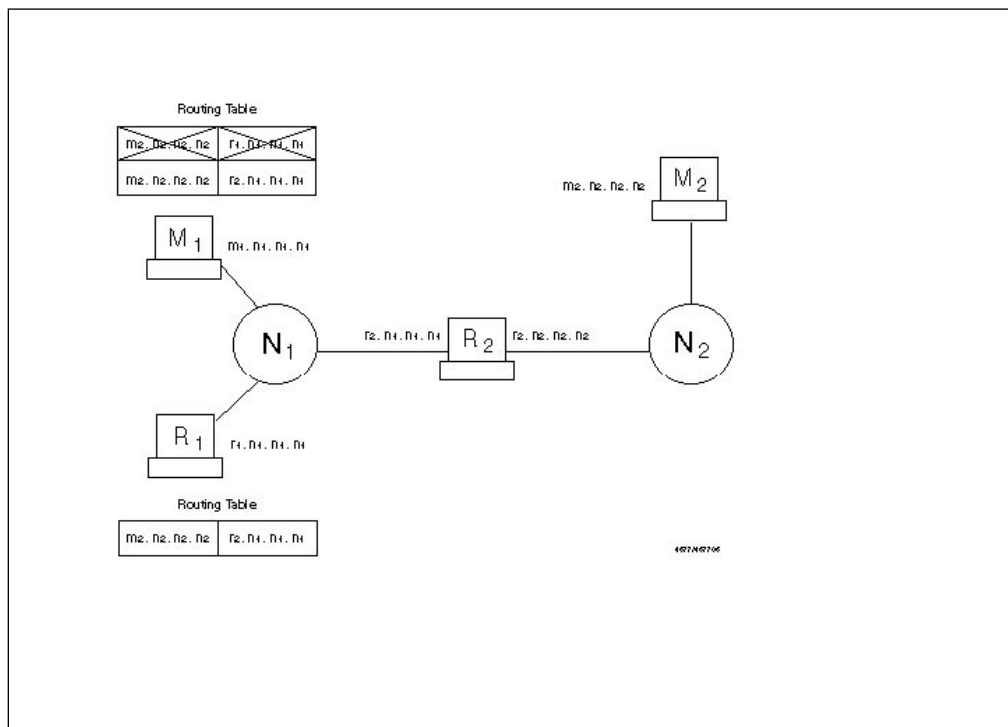


Figure 118. ICMP Redirect, Dynamically Updated Routing Table in M1

Firewall Considerations: Redirect has, as described, a very specific legal use. However it can be abused by a cracker to subvert the routing table and thereby allow IP address spoofing. Redirect is not supposed to cross a router (the packet

is only sent when the sender and both routers are on the same physical network). It may be legal to receive this in the firewall directly if your routing tables are not properly set up. For the same reason, you might allow the firewall to send this type of message.

Our recommendation is to send and log this packet, but not to receive it, as your routing tables should be determined only by you. We also recommend that you notify the owners of the machines to which you sent redirects so that they can correct their routing tables.

7.2.5 Time Exceeded Message

Type	Description	Code
11	Time exceeded	0 - Time To Live Exceeded in Transit 1 - Fragment Reassembly Time Exceeded

Description: Time to live exceeded is generated by a router when it has to forward a packet with a time to live (TTL) value of zero. Fragment reassembly time exceeded is generated by a host when it does not receive all the fragments needed to reassemble a packet.

Firewall Considerations: Enable this for incoming packets so your hosts can perform error recovery. For outgoing packets, allow all fragment reassembly time exceeded messages but not the TTL exceeded messages.

The reason that we recommend blocking TTL exceeded messages from going from the secure network to the non-secure network is that an attacker can use a tool called traceroute to find out which hosts are the routers in your network. This tool manipulates the TTL option of a UDP packet, in order to receive an ICMP TTL exceeded message in response (see 9.15, “Traceroute” on page 202). Blocking the outgoing TTL messages will help you hide your network structure.

7.2.6 Parameter Problem Message

Type	Description	Code
12	Parameter problem	0 - Pointer Indicates the Error 1 - Missing a Required Option 2 - Bad Length

Description: This message is generated when a host that is processing a packet finds a problem in the header parameters that forces the packet to be discarded.

Firewall Considerations: An outsider will gain no information with this packet, so allow it to flow in both directions in order to report problems.

7.2.7 Time Stamp and Time Stamp Reply Message

Type	Description	Code
13	Time stamp message	0 - No code
14	Time stamp reply message	0 - No code

Description: The time stamp message is used to know the time in milliseconds since midnight. It receives as an answer a time stamp reply message.

Firewall Considerations: This protocol may be used by an attacker as a mapping tool (an alternative to ping). We didn't find any reason for allowing it.

7.2.8 Information Request Message

Type	Description	Code
15	Information request message	0 - No code
16	Information reply message	0 - No code

Description: This message is used by a host that is booted across the network to learn in which IP network it is located. It sends an information request packet with both the source and target fields set to zero. The replying host will send the reply with the complete address specified, so the host will now know which IP address it must use.

These messages are obsoleted by new protocols, like RARP, BOOTP and DHCP. Also RFC 1122 says that a host should not implement this protocol.

Firewall Considerations: This message is for local networks only, so it does not need to cross a router. The IBM Firewall should not generate requests, because it knows its IP interfaces, and certainly there is some better place to generate the replies than your firewall, so block it.

7.2.9 Address Mask Request and Address Mask Reply Messages

Type	Description	Code
17	Address mask request	0 - No code
18	Address mask reply	0 - No code

Description: The address mask request message is sent when a node wants to know the address mask of an interface. It expects to receive as an answer an address mask reply message containing the mask of that interface.

Firewall Considerations: This message can be used by outsiders to learn the topology of your network. There were also cases in which a TCP/IP stack took inappropriate actions when it received an unsolicited address mask reply. The address mask request message may be generated by a network management station, such as Netview. Do not allow them in any direction.

7.2.10 Router Advertisement and Router Solicitation Message

Type	Description	Code
9	Router advertisement	0 - No code
10	Router solicitation	0 - No code

Description: These messages are used by hosts in order to dynamically discover the routers in a network. It is specified in RFC 1256, and the current status of the protocol is elective (as listed at the time of writing in the latest RFC of Internet

Official Protocol Standards, RFC 1880). When the host boots, it sends a router solicitation message in order to discover the neighboring routers. The routers reply with router advertisement messages. The router also advertises periodically its routes in an unsolicited way.

Firewall Considerations: These messages are supposed to be for local networks only. They may be received by your firewall, but you should not trust any information they give you. Block these messages.

7.2.11 Domain Name Request and Domain Name Reply Messages

Type	Description	Code
37	Domain name request	0 - No code
38	Domain name reply	0 - No code

Description: These messages are used by hosts in order to learn the domain associated with an address. The host sends a domain name request message and receives as an answer a domain name reply. It is specified in RFC 1788, and the current status of the protocol is experimental.

The idea of this protocol is to substitute the IN-ADDR domain defined in the domain name server (the one that is used in order to translate IP addresses to domain names). Using this protocol, each host will be responsible for the translation of its own IP addresses. The RFC requires every host to implement an ICMP domain name server and also suggests that every host should implement an application for sending the ICMP domain request.

Firewall Considerations: Block it, because it is not currently used.

7.2.12 Traceroute Message

Type	Description	Code
30	Traceroute	0 - Outbound Packet successfully forwarded 1 - No route for Outbound Packet; packet discarded

Description: This message is used in order to implement traceroute (a useful network debugging tool) in a more efficient way. It is specified in RFC 1393, and the current status of the protocol is experimental. The implementation has two parts, a new IP option and the new ICMP traceroute packet.

When a host wants to discover the path to a node, it sends a packet (for example, an ICMP echo request) with the new IP option. Then every router that forwards the packet will also send an ICMP traceroute message to the sender, informing it whether the packet was successfully forwarded or if it was discarded.

Firewall Considerations: Incoming packets, used to trace routes from the secure network to the non-secure network, can be allowed. If you want to hide your internal network structure (you probably should), the outgoing packets must be blocked.

7.2.13 ICMP Message Summary

The following table sums up all considerations we recommend for ICMP message filtering in a firewall.

Table 13. ICMP Message Summary

ICMP Message	Outgoing Packets			Incoming Packets		
	From secure network to secure adapter	From secure network to non-secure network	From non-secure adapter to non-secure network	From non-secure network to non-secure adapter	From non-secure network to secure network	From secure adapter to secure network
Echo	allow	allow	allow	allow	deny	allow
Echo Reply	allow	deny	allow	allow	allow	allow
Destination Unreachable	allow	deny	deny	allow	deny	allow
Source Quench	allow ¹	allow	allow	allow ¹	allow ¹	allow
Redirect	deny	deny	allow ¹	deny	deny	allow ¹
Time Exceeded Code 0	allow	deny	deny	allow	allow	allow
Time Exceeded Code 1	allow	allow	allow	allow	allow	allow
Parameter Problem	allow	allow	allow	allow	allow	allow
Time Stamp	deny	deny	deny	deny	deny	deny
Information Request	deny	deny	deny	deny	deny	deny
Address Mask	deny	deny	deny	deny	deny	deny
Router Ad/Sol	deny	deny	deny	deny	deny	deny
Domain Name	deny	deny	deny	deny	deny	deny
Traceroute	allow	deny	deny	allow	allow	allow

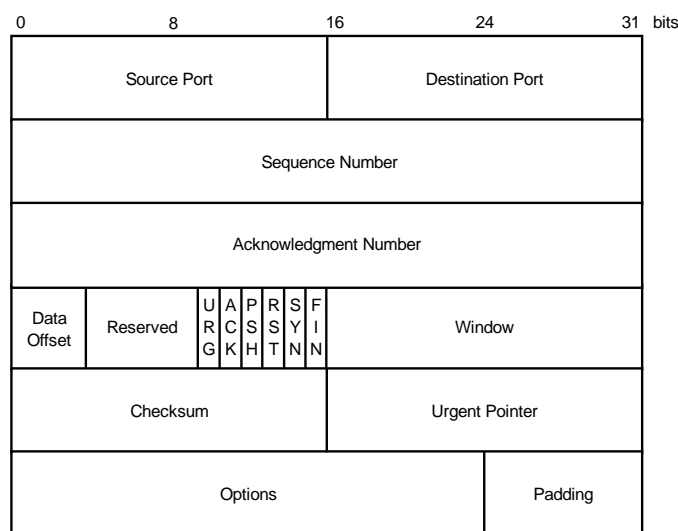
7.3 An Introduction to TCP Packets

TCP is the transport layer protocol that is used by most IP applications. For example, FTP, Telnet, SMTP (mail) and HTTP (World Wide Web) are all higher-layer protocols that use the TCP transport layer. TCP is defined in RFC 793.

¹ Log for later analysis.

TCP provides the application with a reliable end-to-end connection. It takes care of retransmission, lost and duplicate packets and reordering of packets. When a host establishes a connection to another machine using TCP/IP, both hosts will be able to use the same connection in order to send information (that is, it is a two way channel).

Figure 119 shows the format of the TCP header.



5209\PIPP003

Figure 119. The TCP Packet Header

From the firewall point of view, the most important parts of the TCP packet are the source port, destination port and ACK bit. The source port and the destination port are used to identify which process is using a TCP connection. A TCP/IP connection is uniquely defined by:

< Source Address, Source Port, Destination Address, Destination Port >

There are some particular ports that are reserved for specific applications, while others are dynamically allocated for the processes that need them. The reserved ports are normally referred to as *well-known ports*. Appendix E, "Well-Known Port Numbers" on page 613 provides a table with common well-known port numbers. For example, port number 23 is reserved for incoming Telnet connections. Port numbers below 1023 are usually described as *privileged*, meaning that an application needs root authority to use them. This is only a convention, so you cannot base your security policy on it, but one side effect is that the client end of a TCP session normally uses a port number above 1023. We enforce this by policy in our firewall rules.

Let us now see how a TCP connection is established. A TCP session is initiated by a three-way synchronization sequence as shown in Figure 120 on page 140.

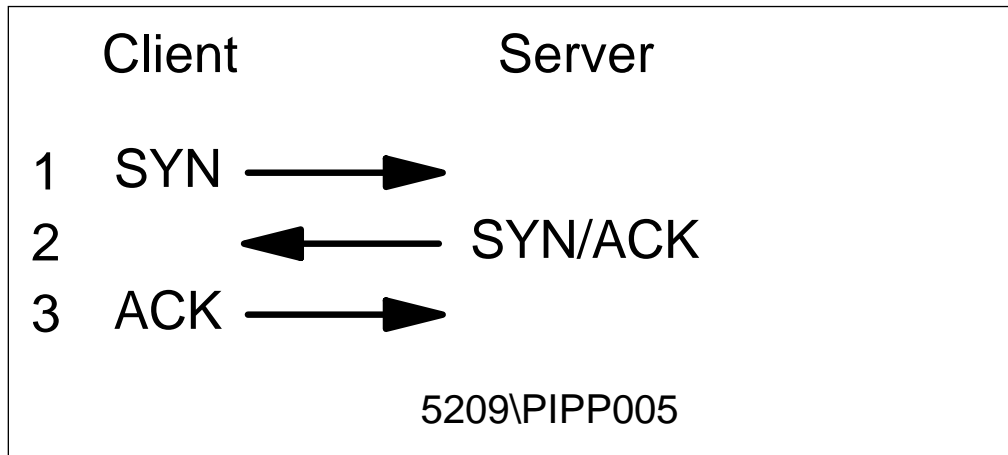


Figure 120. TCP/IP Synchronization Sequence

The acknowledgement (ACK) flag is used by one end of a session to tell the other end that its previous packet was received. The result is that the only packet in the TCP/IP session without the ACK flag set is the SYN packet that creates the session in the first place. So when a connection is created, the first packet does not have the ACK flag set, but all the following packets will have it. Firewall rules use this to control the direction in which a session can flow. If we want to prevent someone from creating connections from the outside (non-secure network) to the inside, we can specify a rule with a protocol specification of *tcp/ack*. This will block the first packet, thereby preventing the establishment of a connection.

This is an effective way to prevent unwanted sessions from being established from outside the secure network. However, for complete security, you should aim to use proxy servers or SOCKS wherever possible.

7.4 Use and Abuse of TCP Ports

If a service normally uses a well-known protocol, that does not mean that it can not use another port. For example, the Telnet server usually uses port 23, but nothing prevents from being run on another port, for example, port 5234.

This must be considered because it might be used to circumvent the firewall restrictions, either by an outsider or an insider. Often, holes in the firewall security are not directly created by attackers, but by unhappy insiders who consider the firewall to be unnecessarily restrictive. An insider that wants to provide an outside access that is not permitted may use a nonstandard port in order to do it. For example, if you prevent your users from providing HTTP servers but allow connections from outside to nonprivileged ports, a user can provide HTTP access using port 5234.

7.4.1 Source Porting

An outside privileged port might be used by an outsider to circumvent your security policy. If, for example, you allow outside access from *tcp/20* (a port usually used by an FTP server for data transfer), an outsider may use this port in order to run another service, for example, a Telnet client. Use of the *tcp/ack* protocol flag can prevent incoming connections, but if you allow an incoming connection from a particular port (as is needed if you want to provide FTP access for your users without implementing a proxy), you will open a security hole in your firewall. We

will come back to this point when we describe firewall definitions for FTP services in 9.4, “FTP: File Transfer Protocol” on page 175.

7.4.2 Stealth Scanning

We have seen how a firewall can control the direction of TCP connections using the ACK bit of the packets. When you want to disable a connection from one direction or the other, you just block the first packet (the one that does not contain the ACK bit), preventing the establishment of the connection.

If an outsider is trying to scan your network in order to discover which machines you have and which services you provide, they will use a port scanner. Usually port scanners try to open a connection to the port. If you use the ACK bit checking in the firewall, this will block the attack.

However, it is possible to scan a network without sending any packet with the SYN bit on. In order to do this, the attacker sends a packet that *looks like* something from the middle of a legitimate session, that is, with the ACK bit on. If the destination port is active, the host will realize that it is not part of a session in progress and send a reset response. If the port is not active, there will be no response. Other types of TCP packets may be used to perform similar types of scanning, such as a packet with SYN, FIN, or ACK in the header or one with the flag field set to 0. All of these packets will be rejected, but the fact that they *are* rejected provides some information about the target machine. This is called *stealth scanning*.

If you want to allow IP forwarding on the firewall and rely on the SYN control, you must be aware that your network might be scanned using these techniques.

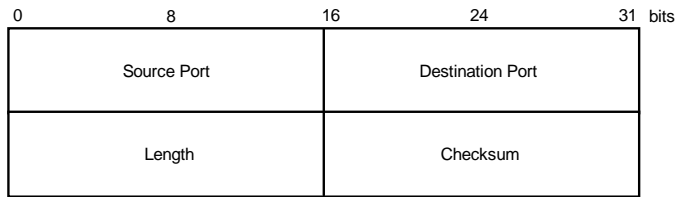
Should you care about this? After all, the attacker cannot establish a useful session this way. The danger is that it provides a way of mapping the contents of your secure network. This knowledge may not be directly useful, but it can become useful if combined with some other back door access. The lesson is this: wherever possible your firewall should be completely dual-homed and not allow any IP routing.

7.5 An Introduction to UDP Packets

UDP, like TCP, is a transport layer protocol, but it is less widely used by applications. Common applications using UDP include the domain name service (DNS) and the simple network management protocol (SNMP). UDP is defined in RFC 768.

Unlike TCP, UDP does not provide the application with a reliable end-to-end connection. Once a UDP packet has been sent, the sender has no knowledge about whether it has arrived or not. It is therefore up to the application to provide acknowledgment and sequence control, if required. UDP is connectionless. That is, each message is a separate entity with no expectation of responses or subsequent request messages. Applications will often mimic the operation of a connection-oriented protocol. For example, a client may use a dynamically allocated port to send a message, and then listen on that same port for a response.

Figure 121 on page 142 shows the format of the UDP header.



5209\PIPP004

Figure 121. The UDP Packet Header

From the firewall point of view, the only important parts of the UDP packet are the source port and destination port. The source port and the destination port are used to identify which process is using a UDP connection. A UDP/IP connection is uniquely defined by:

< Source Address, Source Port, Destination Address, Destination Port >

As in the case of TCP, certain well-known ports are reserved for specific applications. Appendix E, "Well-Known Port Numbers" on page 613 provides a table with common well-known port numbers. For example, DNS uses port 53, and SNMP uses ports 161 and 162.

Because of its connectionless nature, UDP does not have the three-way synchronization sequence of TCP (see Figure 114 on page 132). This means that we do not have the ability to create rules based on the direction in which a session is established. As a result you should avoid routing UDP sessions through the firewall directly. If you do allow them through, you should only allow specific pairs of known endpoints.

Chapter 8. IBM Firewall Rule Base

The control mechanism of the firewall is defined by a rule base. Here we describe the contents of the rule base and work through some simple examples of defining rule base objects. In the next chapter we look at a number of services that you may want your firewall to handle and describe the rule base objects you will need to define.

8.1 Rule Base

The IBM Firewall uses a rule base to determine whether IP packets are passed or blocked at the firewall. The rule base contains a number of components, all related to each other. The main components are called *connections*, each of which exactly defines a specific type of IP traffic to be allowed or prohibited between two network components. Connections are built of one or more *services*. A service is a set of *rules*. A connection also defines the network endpoints (source and destination) to which the services apply. These are defined as *objects*, or *groups* of objects. Figure 122 shows the complete structure.

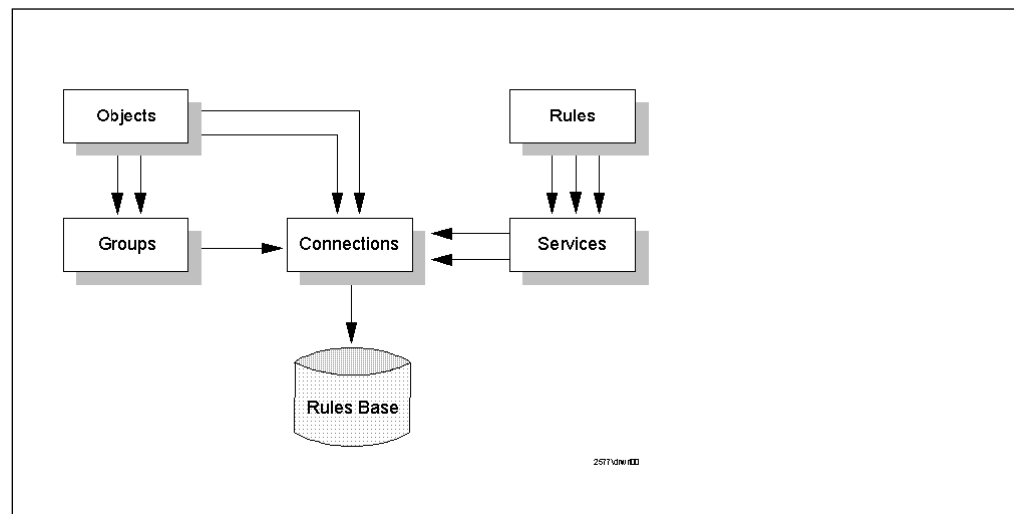


Figure 122. Rule Base Structure

8.1.1 Connections

A connection defines the IP traffic that is allowed or denied between two network components. A connection is built of a source and a destination component, that are connected by a service component, as shown in Figure 123 on page 144. A service defines the type of IP traffic that is permitted or denied between the source and destination. The source and destination in a connection are each defined by an *object* or a *group* of objects.

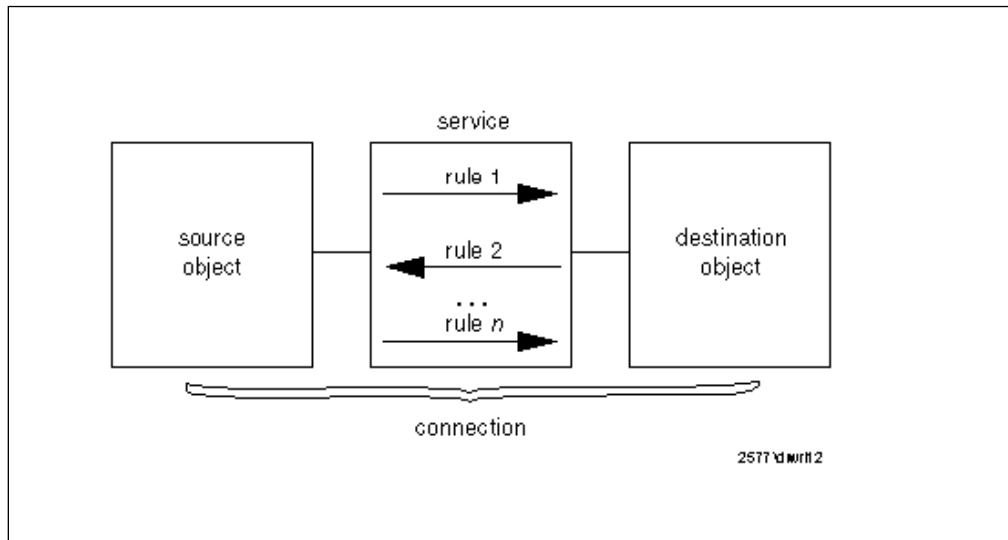


Figure 123. Connection

For example, imagine you have a connection that permits Telnet between a client in the secure network and the proxy server on the firewall. The service in this case is Telnet. To be precise, it is a session from an unprivileged client port to TCP port 23. The source object in this case is any IP address in the secure network, and the destination object is the firewall.

We normally think of a connection definition as something that permits a connection to happen. However they can also be defined to block the defined service. It is most important that the firewall *only* allows the services for which you have permit connections defined. To ensure this happens, it applies a default “block everything” rule to any packet that does not match any rule in a connection definition. Objects, groups and services are described in detail in the following paragraphs.

8.1.2 Objects and Groups

An object is a representation of a network component. It is defined by an IP address and an address mask, so it is possible for one object to represent a whole range of network addresses. A group is a collection of one or more objects. Possible objects are:

- Host: A node in your network with mask 255.255.255.255.
- Network: A set of IP addresses with a specific mask.
- Firewall: The firewall interface with mask 255.255.255.255.
- Router: A unique IP address with mask 255.255.255.255.
- Interface: A network adapter with mask 255.255.255.255.

When you want to define a new object or group, you must select the option **Network Object**, at the left of the main page, of the GUI (see Figure 473 on page 457). For convenience, you can also enter the dialog directly if you find you need to add a new source or destination object when creating a new connection.

The variables in an object definition are:

<i>Table 14. Object Field Meanings</i>	
Description	Meaning
Object Type	One of the possible objects named previously.
Object Name	This is the name of the object. You should try to apply a name convention to keywords used in this name to make it easier to find them again within a list. Don't use the pipe symbol , simple quote ' or double quote ".
Description	This describes the object.
User Name	Use this if you select object type user .
Filter Lifetime	The lifetime of the object rule.
IP Address	The specific IP address or the range of IP addresses for this object.
Subnet Mask	Depends on the type of object you are defining. The subnet mask automatically changes, but you can override it if needed.

The only default object is The World, an object that is matched by any IP address.

8.1.3 Services

A service defines the type of IP traffic that is permitted or denied between a source and destination object. For example, you could construct a service to permit Telnet, or a service to deny Ping.

A service is built of one or more rules. IBM Firewall provides you with a large collection of commonly-required rules and when building a service you can usually find the rules you need predefined. If you don't find the rule that you need, you have to create an extra rule before you define the service. You also have the ability to move rules up or down in the service, to create a specific order in the rules.

In the service configuration window you have the following fields:

<i>Table 15 (Page 1 of 2). Service Field Meanings</i>	
Description	Meaning
Service Name	This is the name of the service. As for objects, you should use a name convention for keywords used in this name, to simplify searching.
Description	This describes the function of the service.
Service Composition	<p>You must add the rules that you need for this service, and you can move rules up or down to establish the correct order of the rules in the service. Order may be important, because some rules contained in a service may be more restrictive than others. If the less restrictive rule is at the top of the rule list, the packet may never be tested against the more restrictive rule.</p> <p>The other element of the service composition section is the <i>flow button</i>. This defines whether the rule applies for packets going from the source to the destination object, or to returning packets (those going from destination to source). Very often a service contains an even number of rules, in pairs, with one of the pair controlling the flow in one direction and the other controlling the reverse direction.</p>

Table 15 (Page 2 of 2). Service Field Meanings

Description	Meaning
Override Log Control	Has the value <i>yes</i> , <i>no</i> or <i>no override</i> . No override (the default) will let the settings in the rules apply. If you select <i>yes</i> a log record will be generated for every packet that matches the rule, regardless of the log control setting in the rules that make up the service. This is useful for debugging, but you will not normally want to log so intensively. If you select <i>no</i> , no log records will be generated regardless of the settings in the rules.
Override Frag. Control	Allows you to override fragmentation settings in the rules. <ul style="list-style-type: none"> • no override - default, rule settings apply • yes - match any IP packet • no - match only non-fragment packets • only - match only non-fragment headers and fragments without a header, do not match non-fragments • headers - match only non-fragments and fragment headers, do not match non-fragments
Time Controls	With this feature you can activate and deactivate the service dependent on the time, date or day of week. You have the following fields available: <ul style="list-style-type: none"> • Control By Time of Day: begin and end time selection in a day. • Control By Days: begin and end day selection for days of the week, or calendar days. • Time Control Action: specify if service is active/inactive during specified period.

When you configure a service you do not specify the objects (that is, network addresses) between which it operates; you define the objects when you place the service in a connection definition. However, you do need to know what type of objects a rule applies to, because you have to define the direction of flow for each rule within the service definition. For example, a service that defines a TCP session from a client to a proxy server on the firewall will only operate as intended if it is included in a connection whose destination object is a firewall IP address.

The flow is indicated by an arrow at the left of the rules, as shown in Figure 124 on page 147.

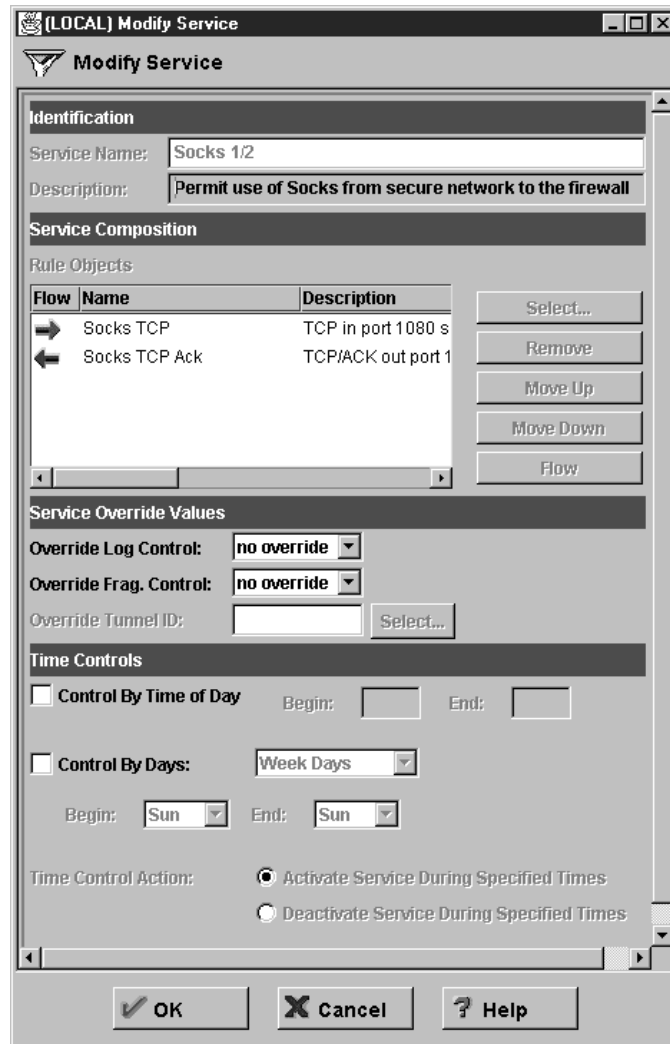


Figure 124. The Flow Indicator

If the rule has a green arrow (arrow points to the right), the filter defined by the rule applies to packets flowing from the source to the destination object. If the rule has a blue arrow (arrow points to the left), the source object and destination object are swapped, so the rule applies to flows from destination to source. For example:

Table 16. Green Arrow

Source Object	The World
Destination Object	Secure Network
Rule	Permit tcp gt 1023 eq 80
Resulting Filter	Permit The World Secure Network tcp gt 1023 eq 80

Table 17. Blue Arrow

Source Object	The World
Destination Object	Secure Network
Rule	Permit tcp gt 1023 eq 80
Resulting Filter	Permit Secure Network The World tcp gt 1023 eq 80

8.1.4 Rules

We have already discussed how rules are combined within services which are, in turn, embedded within connection definitions. Let us now look at rules in more detail.

There are two types of rules: deny rules and permit rules. They define which IP packets are blocked (denied) or allowed to pass (permitted) at which firewall interface. The packets can be filtered on all characteristics described in Chapter 7, "Coming to Grips with IP Packets" on page 127. The fields shown in the following table must be entered when defining a rule.

<i>Table 18 (Page 1 of 3). Filter Rule Field Meanings</i>	
Description	Meaning
Rule Name	This is the name of the rule. Use a name convention for keywords used in this name, so rule names will look consistent. This makes it easier to search for them within a list.
Description	This describes the function of the rule.
Action	Has the value <i>permit</i> or <i>deny</i> . Any IP packet that matches the other fields in the filter definition will either be passed or blocked depending on the value of this field. You can also specify a protocol by number or name.
Protocol	To choose a protocol, open the pull-down menu and select from the following list: <ul style="list-style-type: none"> • all - Any protocol will match this rule. • tcp - The packet protocol must be transmission control protocol (TCP) to match this rule. • tcp/ack - The packet protocol must be TCP with acknowledgement to match this rule. • udp - The packet protocol must be user datagram protocol to match this rule. • icmp - The packet protocol must be internet control message protocol to match this rule. • ospf - The packet protocol must be open shortest path first protocol (ospf) to match this rule. When ospf is specified as the protocol, the source port operation and the source port value is used for the ospf record type value. Filtering can also be performed on the ospf type. A type value of any can be specified and the destination port fields must be specified as any 0. Anything else is ignored. • ipip - The packet protocol must be IP-in-IP protocol (IPIP) to match this rule. When IPIP is specified, the port fields must be specified as any 0.
Numeric Protocol	The numeric protocol allows you to specify a protocol by using its decimal value (according to RFC-1700). Valid values are in the range of 1 to 252. Note that port fields for this rule must be specified as 0 (signifying any port) when using this option. See RFC-1700 for a list of all protocols. Or, you can access the Internet Assigned Numbers Authority (IANA) directly with the Netscape browser.

Table 18 (Page 2 of 3). Filter Rule Field Meanings

Description	Meaning
Source Port/ ICMP Type	The first field specifies the type of operation, the second the desired port number (for ICMP packets it is the ICMP Type of the message). The port operation field is an arithmetic operator field which can have values of: any, equal to, not equal to, less than, greater than, less than or equal to, greater than or equal to. The operator is applied to the desired port field, so, for example, if the two fields were <i>greater than 1023</i> , we would only match packets with a source port number of 1024 or higher.
Destination Port/ ICMP Code	This pair of fields is used in the same way as the source port fields to define which destination port(s) we want the filter to match. For ICMP packets, it refers to the ICMP Code field.
Interfaces Settings	This defines which interface the packet is flowing through: <ul style="list-style-type: none"> • Secure • Non-secure • Specific (a specific interface can be defined, for example, when there is more than one secure interface) • Both (the rule applies when the packet flows through any of the firewall interfaces)
Direction/ Control	In some cases the firewall may act as a router, in which case packets flow through it. In other cases the packets may go to an application on the firewall machine itself (such as a proxy server). This field defines whether the packet has a destination or source of the firewall, or whether the destination and source are both addresses other than the firewall (in which case the firewall is behaving as an IP router). Possible values are: <ul style="list-style-type: none"> • Local (coming to or from the firewall itself) • Route (going through the firewall) • Both (the rule applies in all cases)
Direction	Defines whether the packet is coming into or going out of the adapter where the rule is applied. Remember that the rule can apply at any of the firewall adapters, controlled by the Interface definition (above). Possible values for the direction are: <ul style="list-style-type: none"> • Inbound • Outbound • Both (rule applies whichever way the packet is going)
Log Control	This field defines if the packet should be logged or not. The default log control setting for permitted packets (those that pass the rule) is <i>no</i> and for denied packets is <i>yes</i> . It is important to log intensively on a firewall, because you cannot tell in advance which piece of seemingly unimportant log data will reveal an attack. However, logging every successfully transmitted packet is usually more than you need. Options are: <ul style="list-style-type: none"> • No • Yes

Description	Meaning
Fragmentation Control	<p>The possibilities are:</p> <ul style="list-style-type: none"> • Yes - matches header, fragments and nonfragmented packets • No - matches only nonfragmented packets • Only - matches only fragment headers and fragments without a header • Headers - match only non-fragments and fragment headers, do not match non-fragments

8.1.5 Names and Descriptions Convention

In this section we explain the reasons why we suggest you should use a convention when you define a name for a new object, rule, service or connection and its description.

Three characteristics are important when you design a name or description:

- Identify

When you add a new definition (connection, service, rule or object) to the rule base it joins the large list of pre-loaded definitions. You need to be able to identify easily the purpose and function of the new definition. A short name and a good description can help you to do it.

- Reuse

When you need to define a connection or service, it is most easy to create this service using existing rule templates. So if you create the rules with easy names, you can more easily select the appropriate rules for new services.

- Change

Reuse of definitions is certainly a good thing, but caution is needed. Sometimes it may be better if you create a new rule for a special service, instead of re-using an existing rule, because when you change a rule used by multiple services, the change will affect all the services, which may not be what you want.

We give some suggested rules for a naming convention which you may want to follow, or use as a basis for your own. For the names we use only two or three words, and for the description we explain the name better. We place the most descriptive word first in the name, so that all of the definitions that are related to each other appear together when they are presented in an alphabetical list. For example:

	Name	Description
Object	Firewall secure	This is the secure firewall interface
Rule	TCP-80 permit inbound	Permit TCP port 80 secure inbound
Service and Connections	HTTP outbound permit	Permit HTTP from secure network to non-secure network

Remember don't use the pipe symbol |, quote ', or double quote " in the names and description. Also, remember that you can see any of the rule elements as fields in the rules list; you just need to click the button labelled with the name of the field that you need. For example, in Figure 132 on page 156 we could add the Direction field to the rule list display by clicking on the button at the top of the list.

8.2 Rule Base Design

To set up the rule base for your firewall in a structured way it is important that you have a clear picture of your network infrastructure and the services that you want to provide. In this way it is easy to configure your firewall and maintain a consistent set of connections, objects, services and rules.

When you want to implement your connections there are basically two possible types of connections. The first type, and most easy to implement, is a standard connection. This connection can be built with predefined services. The following services are predefined:

- Telnet
- FTP
- HTTP
- SOCKS
- SSL
- SMTP
- Identd
- SNMP
- Ping
- DNS
- SecureID
- Remote Logging
- Firewall Configuration

The other type of connection cannot be built of predefined services and you must define your own service instead. When defining the service you will probably have to define your own rules as well. Note that some protocols have more than one predefined service and that sometimes a predefined service does not contain the exact rules you need.

We will first describe how to set up a standard connection (one made up of predefined services and rules) and after that a non-standard connection.

8.2.1 Standard Connections

In this part we explain how to define a connection in the IBM Firewall. The example we use is a Telnet connection between the secure network and the firewall secure interface. This is something you are very likely to need to set up, to allow an administrator to log in to the firewall for maintenance purposes.

First, start the Configuration Client from the Windows NT desktop by selecting **Start** → **Programs** → **IBM Firewall** → **Configuration Client** and then select the **Connection** option from the Navigator panel on the left of the display. A list of existing connections will appear, as shown in Figure 125 on page 152. In the list select **NEW** and click on **Open** to create a new connection.

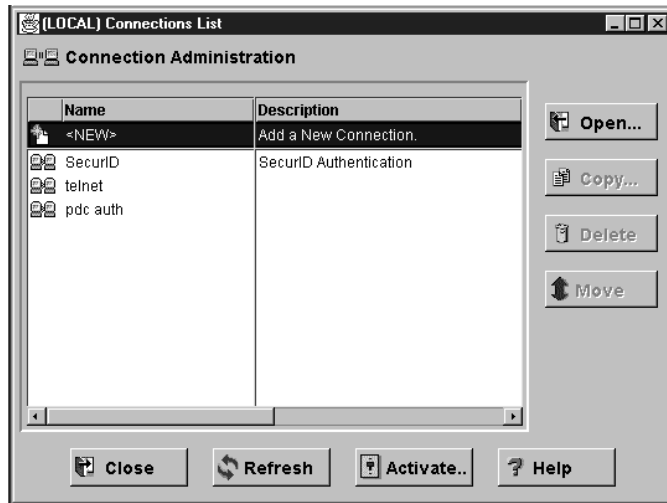


Figure 125. Connection List

Figure 126 shows the new connection screen, where you must define all the parameters. First, enter the name and the description of this connection. Remember to use a convention for all the names, as this will make future definitions and modifications easier.

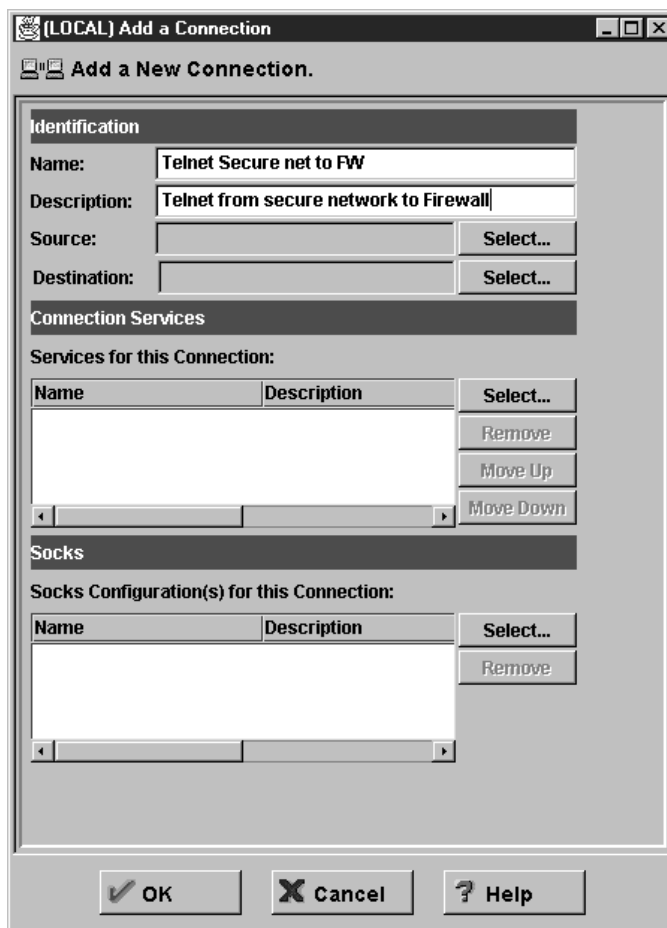


Figure 126. Add New Connection

Secondly you need to define the source object and destination object. Click on **Select** to select each one from the object list. If you have not already defined the object, you can select **New** to define it. The only object predefined is The World, so we will have to define both the source and destination objects to construct our example. Figure 127 shows the definition for our source object, representing any address in the secure network. After you have defined the object click **OK**, then select the new object in the object list and click **OK** to place the object in the source object field. This procedure must be repeated for the destination object, the firewall itself. Figure 128 on page 154 shows our definition for this object.

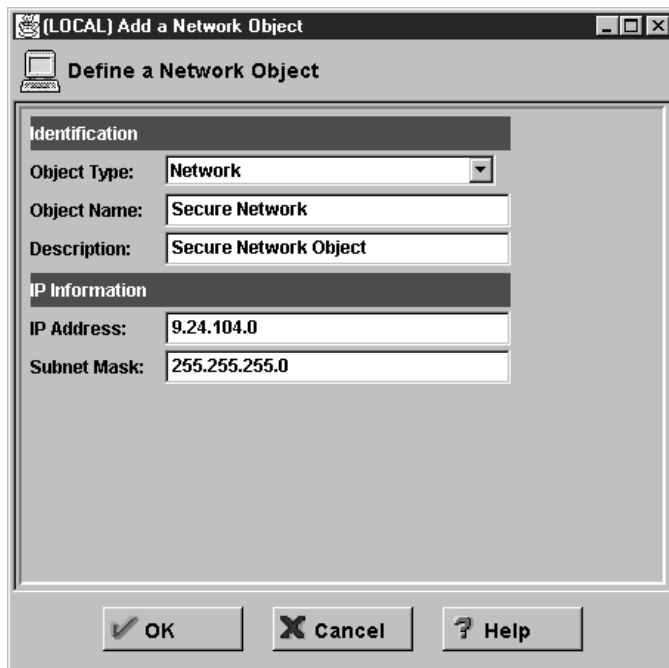


Figure 127. Source Object

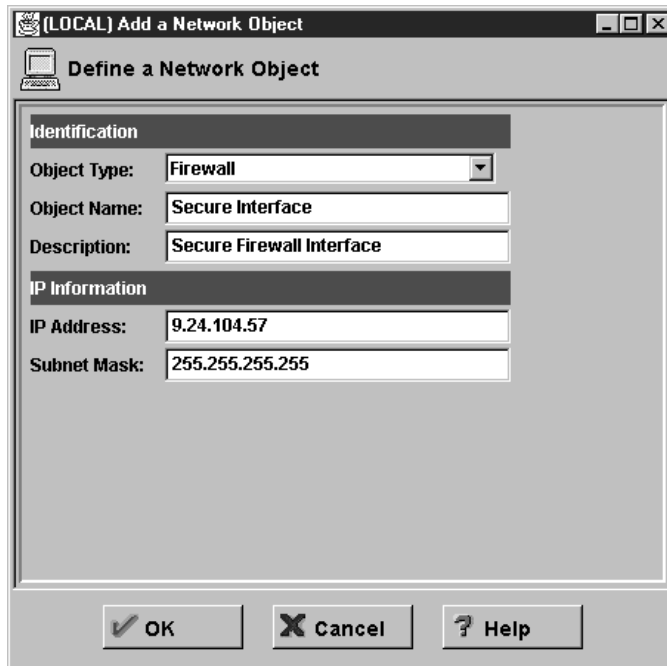


Figure 128. Destination Object

Finally you need to select the service between these objects. Click on **Select** and a list of all the defined services will appear. In this case we are using a standard service, so select **Permit Proxy Telnet Outbound** and click on **OK**. Figure 129 shows the services list.

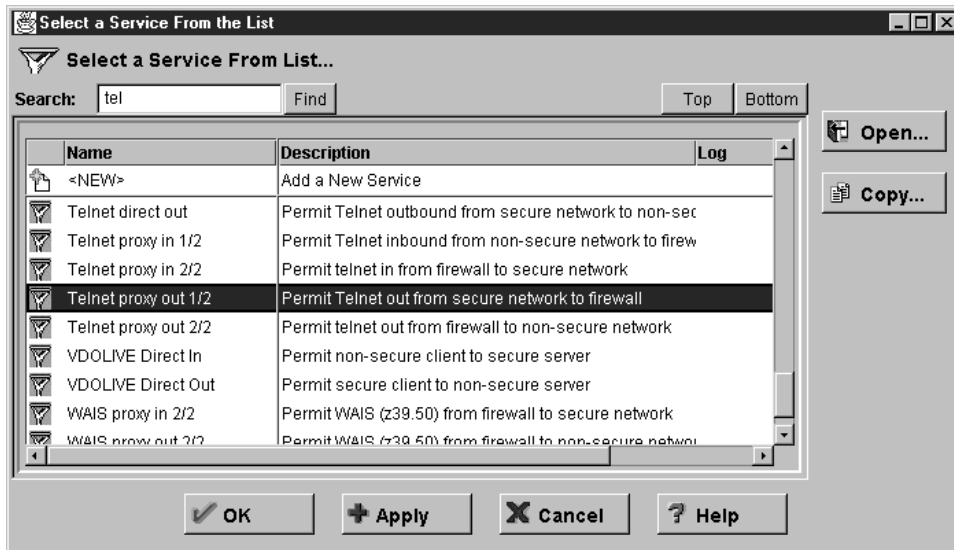


Figure 129. Services List

Figure 130 on page 155 shows the final result. Click on **OK** to save the connection definition.

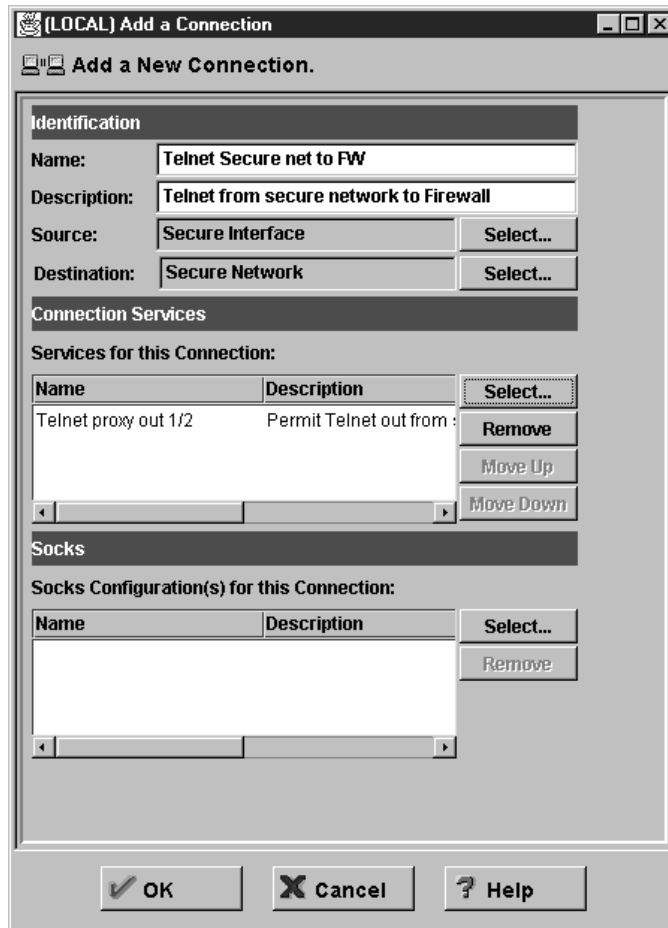


Figure 130. Connection

Now you need to activate the rule base and the filters file will be rebuilt. We describe this in 8.2.3, “Rule Base Activation” on page 159.

8.2.2 Non-Standard Connections

A non-standard connection is one that cannot be built from predefined services. For example, imagine you have a new application (we call it *CUST*) which has a proxy server running on your firewall. It listens on TCP port 400 and you want to be able to access it from the secure network. This is visualized in Figure 131.

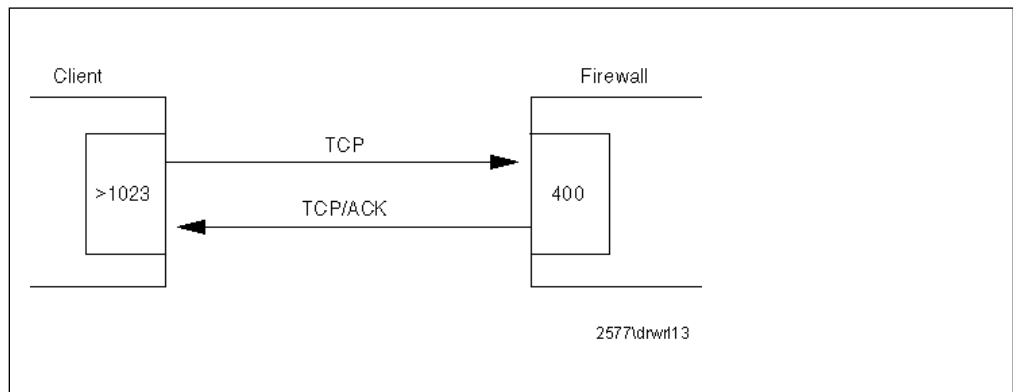


Figure 131. Non-Standard Connection

To be able to build this connection we are going to create a service that is called *Permit CUST*. First, we have to decide whether we need new rules for this service. Therefore, you have to know which rules already exist by checking the list of rules. Do this by selecting **Traffic Control** then **Connection Templates**, and **Rules** from the initial GUI navigator panel (see Figure 473 on page 457).

In this example we need a rule that permits inbound TCP packets on port 400 of the secure interface. This does not exist so we must create a new rule.

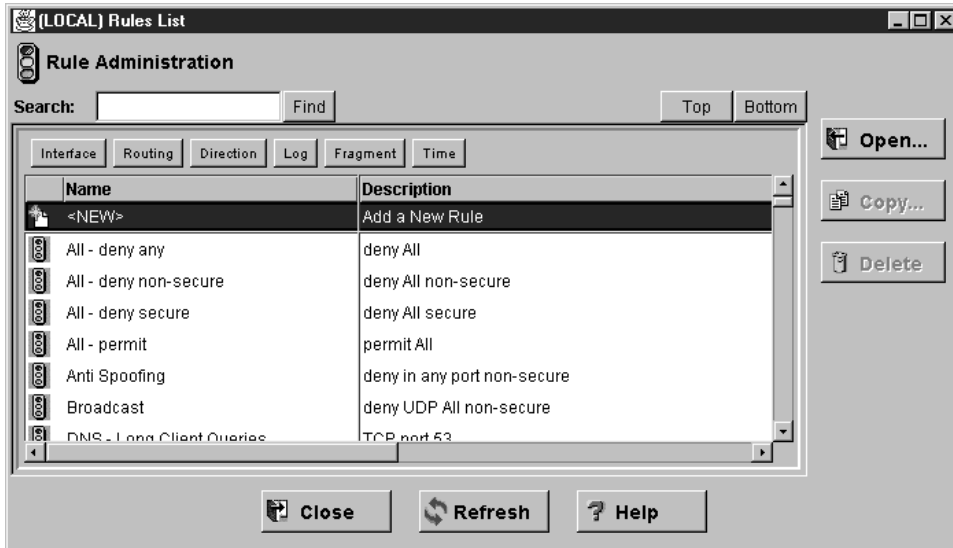


Figure 132. Define a New Rule

It is very important to assign a clear name to a rule. For example, do not use the name of a source or destination in the rule name, because they are independent of the rule. A good name may be *Permit CUST Inbound 1*. By giving rules clear names, it is also easier to reuse your rules. In the rule list double-click on **<NEW>** as shown in Figure 132. Fill in the parameters for the new rule, as shown in Figure 133 on page 157. Notice that we have been very specific in defining the rule: it will only allow packets for our CUST application to pass if they appear inbound on the secure side of the firewall.

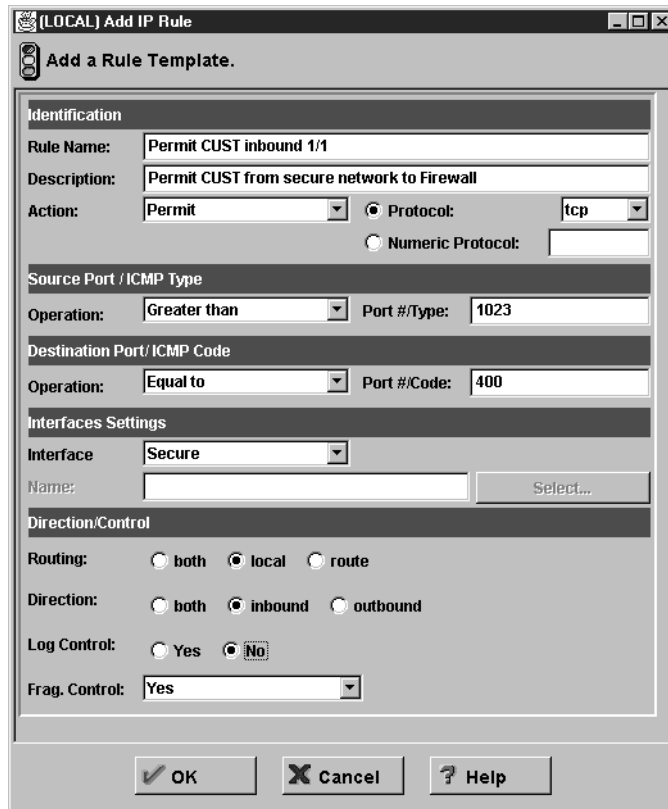


Figure 133. Parameters for Permit CUST Inbound 1/2

This rule only deals with one direction, client to server. We also need to create a rule for the response packets from the server to the client. The construction of this rule is shown in Figure 134 on page 158. The differences from the first rule are:

- The protocol is now TCP/ACK.
- Source and destination criteria are swapped.
- The direction is now outbound.

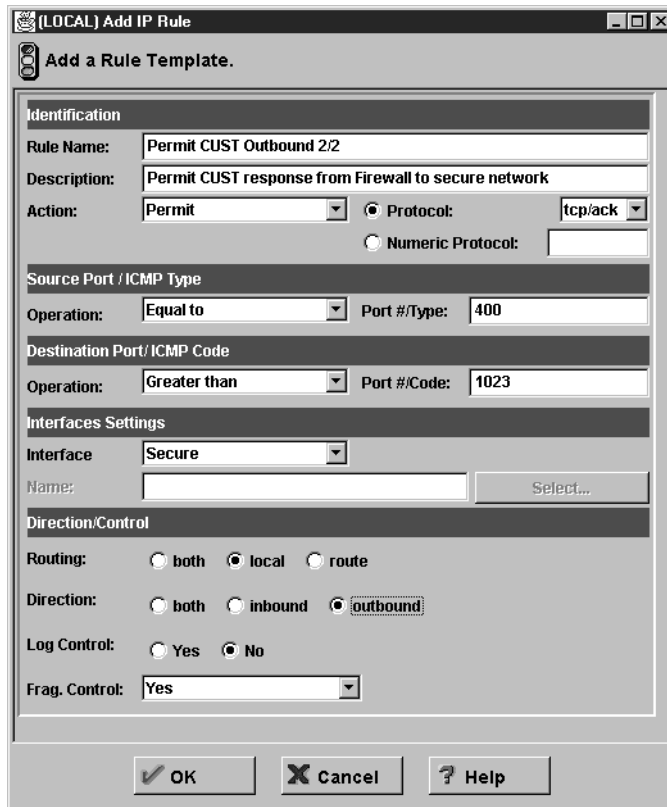


Figure 134. Parameters for Permit CUST Outbound 2/2

After creating this rule we can build the *Permit CUST* service that invokes the new rules. Select **Traffic Control** then **Connection Templates**, and **Services**. from the initial GUI navigator panel (see Figure 473 on page 457). You will see the list of existing services. In the list double-click on **<NEW>** to see the new service dialog screen shown in Figure 135 on page 159.

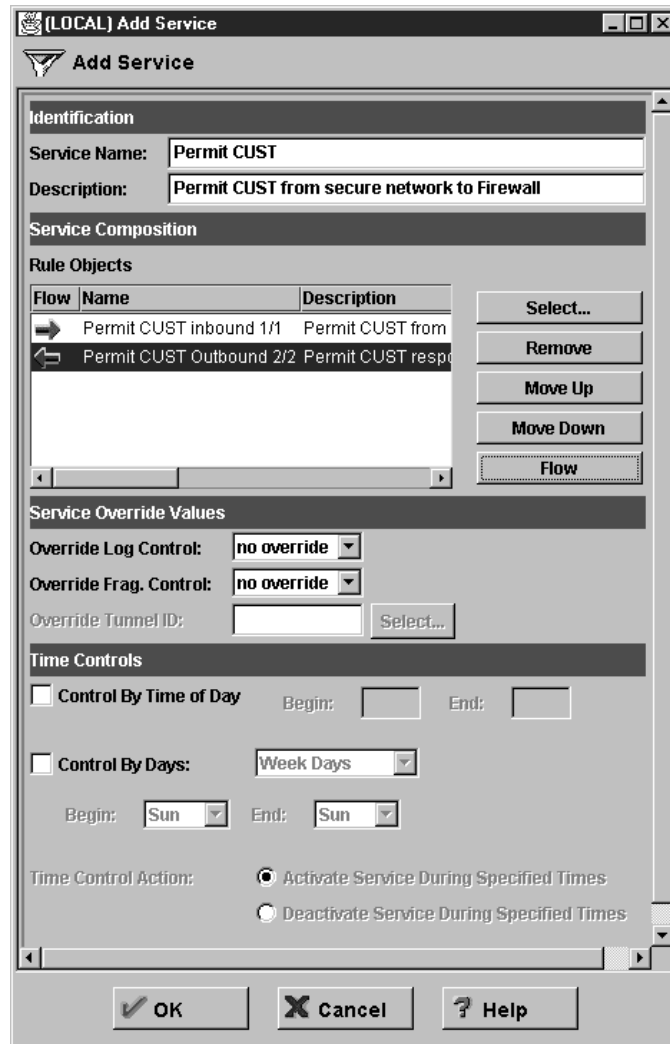


Figure 135. Create Service Permit CUST

Notice that the flow of the second rule has to be changed, because it applies to packets in the reverse direction (destination to source).

Finally, we can configure a connection, in the same way as for the previous example (see 8.2.1, “Standard Connections” on page 151) with the following content:

- Source object *Secure Network* (created in the previous example).
- Destination object *Secure Firewall* (also created in the previous example).
- Service *Permit CUST*.

8.2.3 Rule Base Activation

After you have defined the rule base you have to activate it. This will also create the filters file that IBM Firewall uses to control its packet filtering function (\Program Files\ibm\Firewall\config\fwfilters.cfg).

Figure 136 on page 160 shows the options in the rule base activation window. The following list describes these options:

- Regenerate Connection Rules and Activate. If you choose this option, the filters file will be created, using your definitions of connections, objects, rules, services and SOCKS. Packet filtering will be done according to the filters file. You cannot edit the filters file manually, because every time that you activate the rule base, the file fwfilters.cfg is overwritten.
 - Deactivate Connection Rules. If you choose this option, your rule base will be deactivated and packet filtering will be disabled.
- Note:** The firewall will not route packets if the connection rules are disabled.
- List Current Connection Rules. If you choose this option, you will see the content of the filters file \Program Files\ibm\Firewall\config\fwfilters.cfg.
 - Validate Rule Generation. If you choose this option the syntax of the rules in the rule base will be validated.
 - Enable Connection Rules Logging. Choose this option to enable logging.
 - Disable Connection Rules Logging. Choose this option if you don't want logging. This is not recommended!

Figure 136 shows the activation screen.

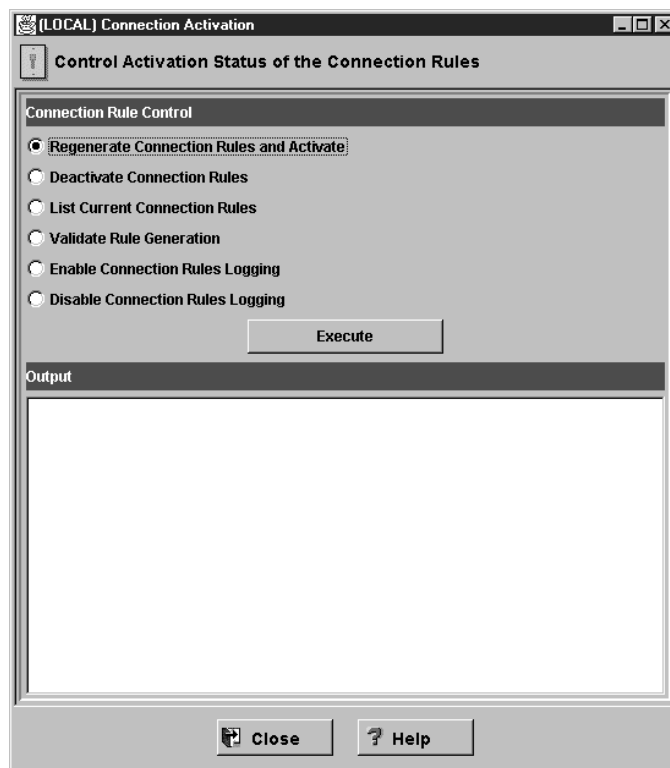


Figure 136. Connection Activation

8.2.4 Performance

A lot of people worry about how fast a firewall can handle the rules configuration. The more active filter rules we have, the more slowly our system will be. However, in most firewall implementations, the limiting factor on performance is associated to the speed of the link to the Internet, not to the firewall itself.

Chapter 9. Examples of Rules for Specific Services

The objective of an Internet connection is to provide access to *services*:

- To allow your users to access services in the Internet
- To allow Internet users access to services that you provide

In most cases a firewall is configured to allow access to a combination of services of different types. In this section we will analyze some of the different services that you will want to provide, such as terminal emulation, file transfer or World Wide Web. We will consider the most important alternative ways to deliver them and give examples of rules to control them.

For a more extensive discussion about all the possible services, we recommend referring to *Building Internet Firewalls* (Chapman and Zwicky).

9.1 What Services Should You Provide?

One of the most important points is to decide which services you will provide. You should provide only those services that your users *need*, not the ones that they merely *want*. This point cannot be stressed enough. A service should be provided only if there is a business requirement, not if it's a "nice to have" feature. Therefore, a service should always comply with corporate's security policy.

It is good practice to use deny rules at the end of each section of rules that permit a given service, instead of relying on the implicit default deny everything rule (see Chapter 8, "IBM Firewall Rule Base" on page 143). This will prevent problems that may arise if the rules file contains a later misconfigured permit rule.

In most of the examples below, we show two diagrams. The first is a schematic diagram of the network connection we are trying to achieve. The second represents the combination of rules, services and objects that make up the connection definition in the firewall rule base.

The schematic diagrams contain IP address information that is specific to a given installation. In order to make the samples in this chapter as generic as possible, we have substituted symbolic notations for IP addresses, masks and functions. The following list summarizes these notations:

s.s.s.s	Secure network IP address
sm.sm.sm.sm	Network mask of secure network
s.s.s.1	IP address of firewall secure interface
n.n.n.1	IP address of non-secure firewall interface
r.r.r.r	Remote Secure network IP address
rm.rm.rm.rm	Network mask of remote secure network
t.t.t.1	IP address of firewall non-secure interface
t.t.t.2	IP address of remote firewall non-secure interface
M.M.M.M	Internal mail server IP address
SN	Abbreviation for secure network
NSN	Abbreviation for non-secure network
N.V.6.K	Network manager (using SNMP)
S.Y.S.L	syslog host (destination of logging)
D.N.S.I	Internal DNS

N.S.E.R Internal news server
N.F.E.E External news feeder
p.p.p.p Proxy
PPP1, PPP2, etc Generic port numbers

The connection diagrams use a convention to illustrate the relationship between the different rule base components, as shown in Figure 137.

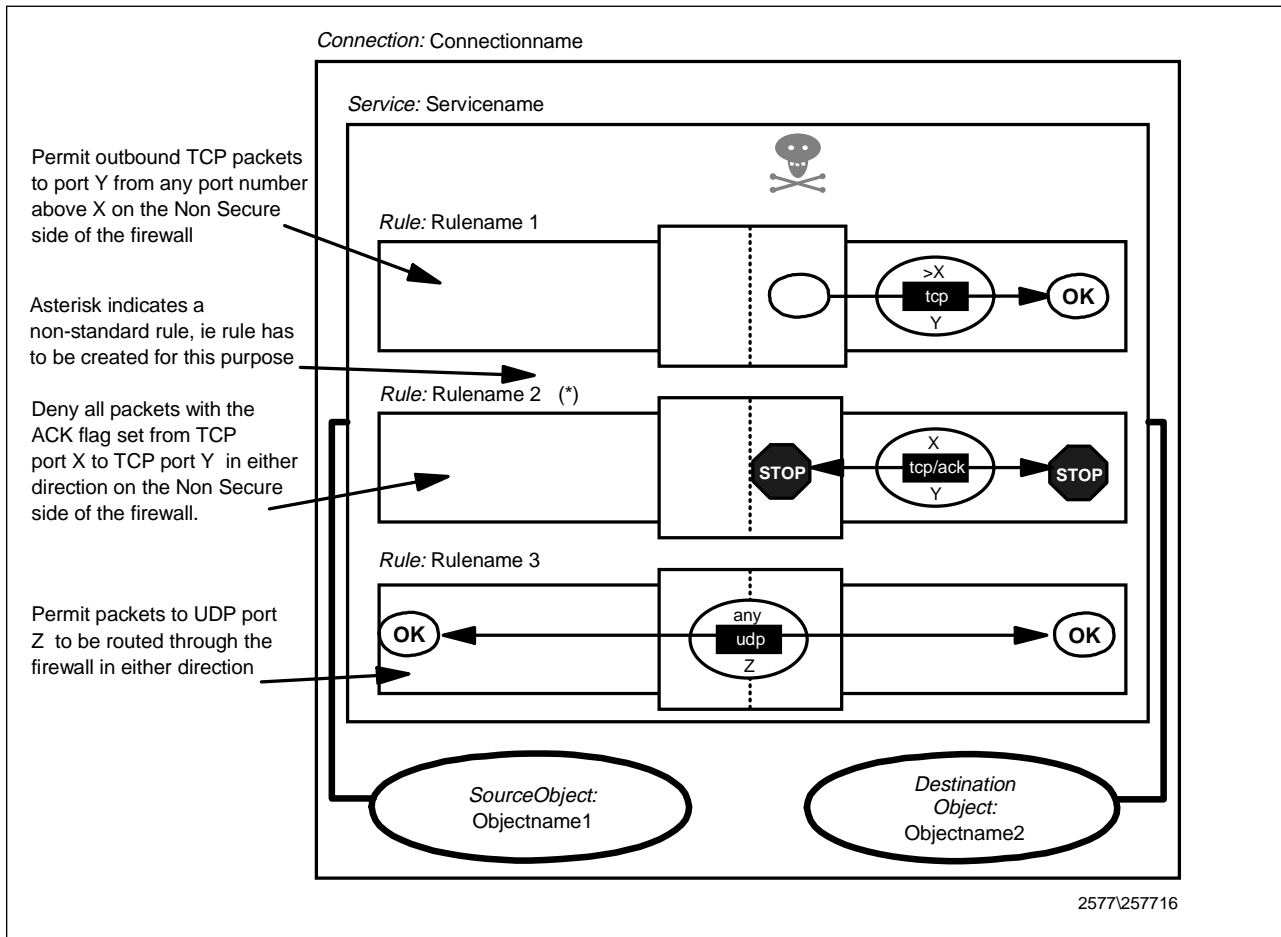


Figure 137. Key to the Connection Diagrams

9.2 Connection Rules that Should Always Be Present

No matter what services you are offering, you should always start your rule base with a set of rules that provide general protection, by explicitly denying several types of attack:

- Rules to block attempts at IP address spoofing
- Rules to control ICMP message flow
- A rule to isolate private networks from the Internet
- A rule to protect the SOCKS service on the non-secure interface
- A rule to protect from misuse of the loopback network

You may also wish to place other rules near the top of the rule base, for example, to control broadcasts or prevent routed traffic. We will consider each of these in turn.

9.2.1 Rules to Block Attempts at IP Address Spoofing

In an IP address spoofing attack, the attacker impersonates the IP address of another machine. This is typically used to attack services that rely on IP addresses for user authentication. This attack is different from the source-routing attack, although many people confuse them. In both cases the attacker sends packets that appear to come from a trusted machine. The difference is that in attacks which subvert IP routing, the attacker expects also to receive packets that are intended for the trusted machine, whereas in the IP spoofing attack, the attacker sends packets that are harmful enough to do damage without the need for a direct reply. In this case, the attacker doesn't need to receive any IP packet; it is enough to be able to send packets.

A classic attack of this type is the TCP sequence number prediction attack. In this attack, the attacker does not need to receive any packet from the attacked host. It needs to be able to send packets and it also has to predict the TCP sequence number that will be used by the attacked machine in its replies to the impersonated machine (so that it can acknowledge them).

Figure 138 illustrates this. In this example, we have a truster machine that trusts a machine with IP address 9.24.104.241. The attacker wants to send packets with the trusted address as source, instead of its own. To do this the attacker has to do the following three things:

1. It must somehow cause the real trusted machine to not respond to the truster. Any denial of service attack could be used, including the Ping 'O Death and SYN flooding attacks. Alternatively the attacker could just wait until the machine was down for some legitimate reason.
2. Next it sends TCP/IP packets to the truster, constructed with a source address of the trusted in the header. The truster will respond in the normal way, believing that it is receiving requests from the real trusted.
3. The responses have nowhere to go, and normally the timeout processing on truster would detect that trusted was not responding. In order to maintain the illusion, therefore, the attacker must be able to predict the sequence numbers of the packets that the truster sends. It can then fabricate requests and responses which appear to acknowledge the truster's messages.

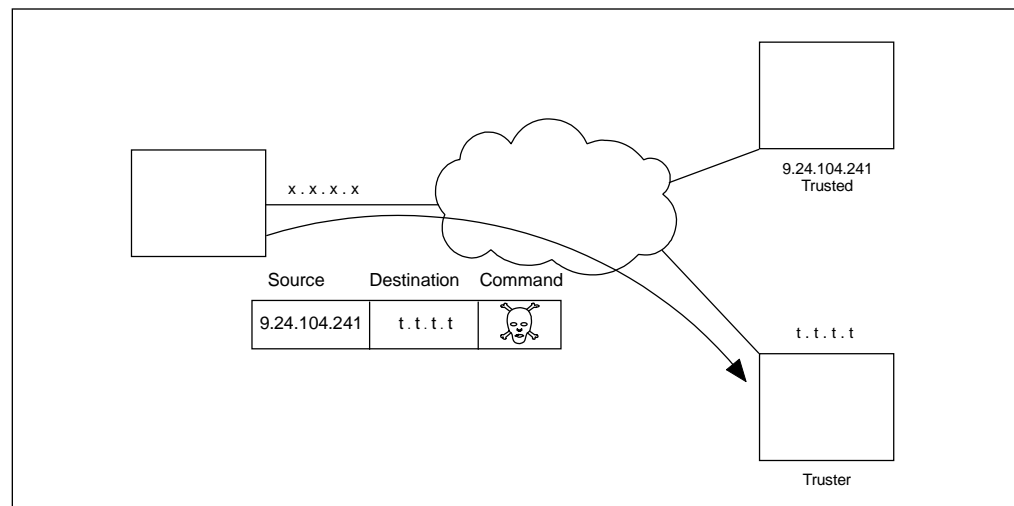
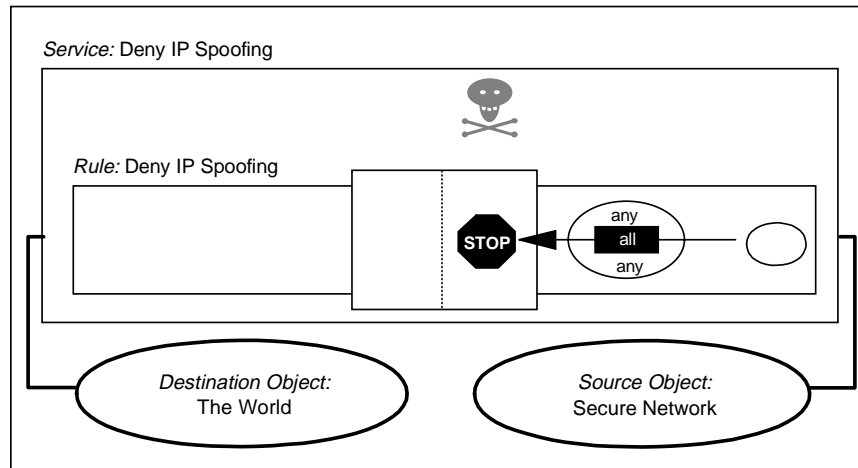


Figure 138. IP Spoofing Attack

To protect your network from address spoofing and routing corruption attacks, you should add a couple of connections to reject packets that have a source address within the secure network, but which appear on the non-secure interface of the firewall (see *Actually Useful Internet Security Techniques* by Larry Hughes for a more complete discussion).

The following connections implement this policy. These should be the first two connections in your rule base.

Connection: Block Inbound Spoofing



Connection: Block Outbound Spoofing

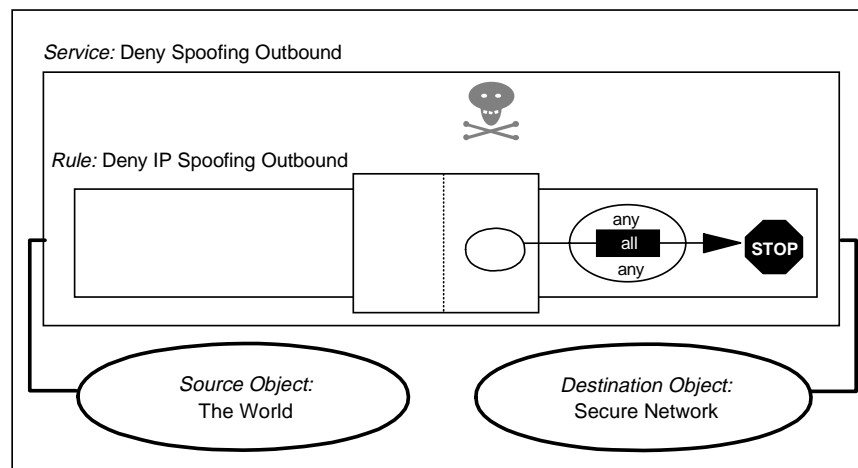


Figure 139. Connection Definitions to Prevent IP Spoofing

The first rule used in these connections is part of the standard rule base, so you must create the second one and add it to the *Anti Spoofing* service. Or add another service with the second one. See Chapter 8, "IBM Firewall Rule Base" on page 143 for discussion about connection, services and rules.

The filter definitions resulting from these connections are:

```
deny s.s.s.s sm.sm.sm.sm 0 0 all any 0 any 0 nonsecure both inbound
deny 0 0 s.s.s.s sm.sm.sm.sm all any 0 any 0 nonsecure both outbound
```

9.2.2 Rules to Control ICMP Message Flow

The simplest thing to do is to block all ICMP messages from crossing the firewall. A simple connection to do this would be as follows:

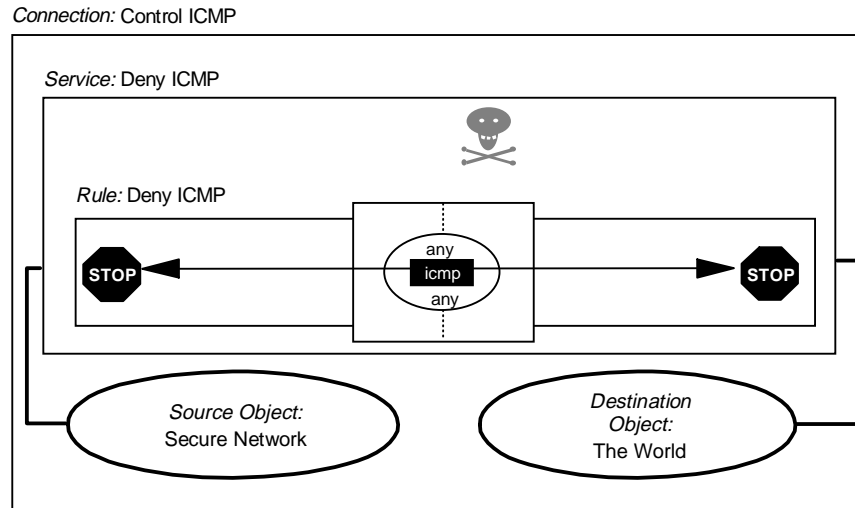


Figure 140. Connection Definition to Block All ICMP Messages

However, if you want to be more selective about which ICMP messages you want to allow to flow, you may want to add the connections described in 9.20, "Filtering Specific ICMP Messages" on page 216.

The filter definition resulting from this connection is:

```
deny 0 0 0 0 icmp any 0 any 0 both both both
```

9.2.3 Rule to Isolate Private Networks from the Internet

RFC 1627 defines ranges of IP addresses that are reserved for private, isolated networks. What this means is that the address ranges can be used within an organization, but they can never be one end of a session that crosses the Internet. The Internet backbone routers are configured not to route them.

You should implement the following connections to prevent these addresses from leaking into or out of your network:

Connection: Isolate Private Networks

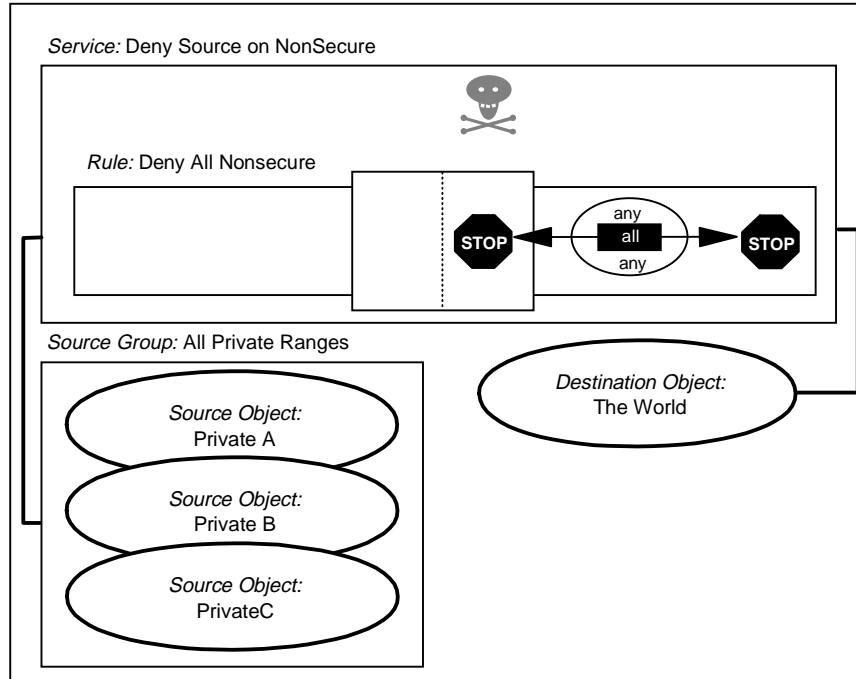


Figure 141. Connection Definition to Control RFC 1918 Addresses

The three network objects in this connection, which are collected together in the *All Private Ranges* group are all the nodes in the ranges defined in RFC 1918. That is to say, the following:

```
Private A: Address = 10.0.0.0    Mask = 255.0.0.0
Private B: Address = 172.16.0.0  Mask = 255.240.0.0
Private C: Address = 192.168.0.0 Mask = 255.255.0.0
```

You may think that you do not have these addresses in your network, but in practice they are often used for testing purposes, so they may appear without warning.

The filter definitions resulting from this connection are:

```
deny 10.0.0.0 0xff000000 0 0 all any 0 any 0 nonsecure both both
deny 172.16.0.0 0xffff0000 0 0 all any 0 any 0 nonsecure both both
deny 192.168.0.0 0xffff0000 0 0 all any 0 any 0 nonsecure both both
```

9.2.4 Rule to Protect the SOCKS Service on the Non-secure Interface

The SOCKS server normally only operates inside-out. That is, it provides a facility to allow secure network clients to access non-secure servers, but not vice versa. You should therefore place an explicit connection that prevents a non-secure node from connecting to it:

Connection: Protect Socks

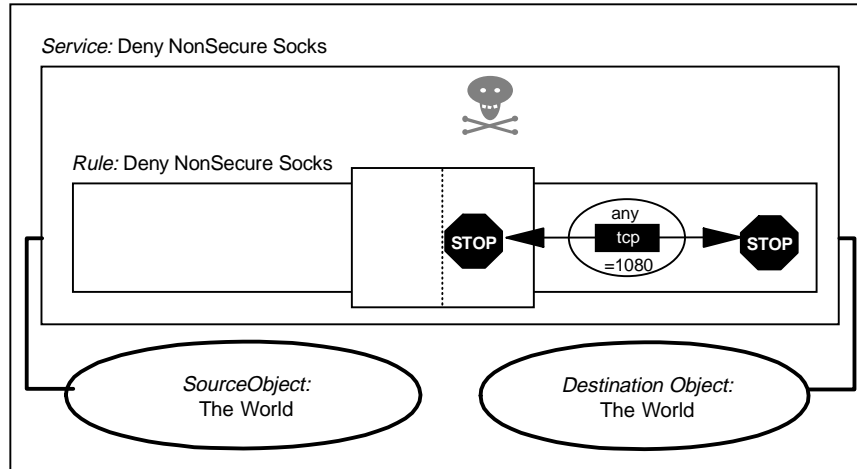


Figure 142. Connection Definitions to Protect SOCKS

The filter definition resulting from this connection is:

```
deny 0 0 0 0 tcp any 0 eq 1080 nonsecure both both
```

This rule can also be set by using the Security Policy panel (see 9.23, "Using the Security Policy Panel" on page 219).

In the current version of IBM eNetwork Firewall for Windows NT, SOCKS V5 is supported. One main feature of SOCKS V5 is authentication. Therefore, it is possible to allow incoming SOCKS connection from outside. If you do that, you should use a strong authentication scheme, like SecurID. We tried an incoming SOCKS connection in our lab to test Host on Demand (HOD) from non-secure network. It is discussed in 9.19, "HOD: Host On-Demand" on page 210. For more information on SOCKS please refer to Chapter 13, "Configuring Proxy Services and SOCKS" on page 293.

So you should not implement this rule if you plan to allow incoming SOCKS.

9.2.5 Rules to Protect from Loopback Network

Loopback is a logical IP interface that IP uses for internal communications. The loopback addresses should never appear on any real network interface. The following connection makes sure that they don't:

Connection: Protect From Loopback

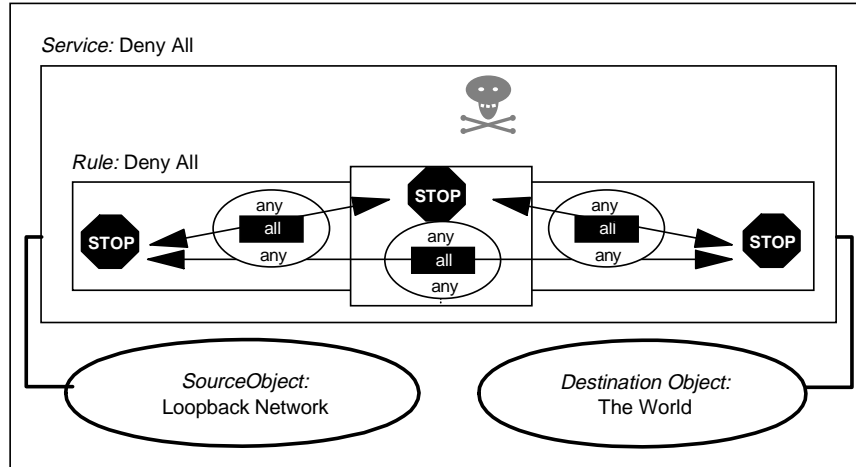


Figure 143. Connection Definition to Protect Loopback

The filter definition resulting from this connection is:

```
deny 127.0.0.0 0xff000000 0 0 all any 0 any 0 both both both
```

9.2.6 Broadcast

Broadcasts should be blocked in order to reduce network traffic on the firewall. You also should be aware of multicast traffic, for example, the *all host multicast* address (224.0.0.1). See RFC 1112 *Requirements for Internet Hosts -- Communications Layers* for information about IGMP (Internet Group Management Protocol) and multicasting. Denying broadcast and multicast packets can cause excessive logging, so we recommend that you switch logging off for these *denial services*.

Connection: Deny Broadcasts on NonSecure

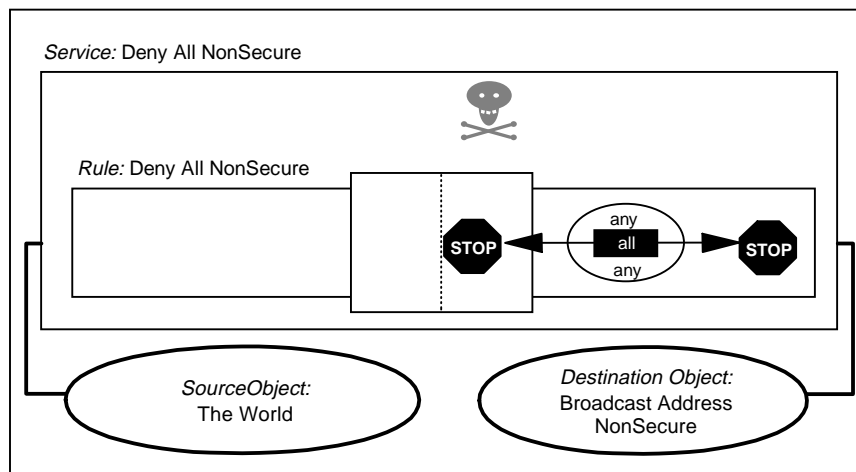


Figure 144. Connection Definition to Block Broadcasts

A similar connection can be implemented to reduce traffic on the secure interface.

The filter definitions resulting from this connection are:

```
deny 0 0 224.0.0.1 0xffffffff all any 0 any 0 nonsecure both both
deny 0 0 255.255.255.255 0xffffffff all any 0 any 0 nonsecure both both
```

The second rule only stops broadcast to 255.255.255.255; you may want to modify it for your specific case. You should use the inverse of the subnet mask used for the non-secure interface as the broadcast address to stop.

Similar rules can also be set by using the Security Policy panel (see 9.23, “Using the Security Policy Panel” on page 219). The rule created when you check **Deny broadcast to nonsecure interface** on the Security Policy panel is:

```
deny 0 0 0.0.0.255 0.0.0.255 udp any 0 any nonsecure both both
```

This rule works for broadcasts to class C addresses only.

9.2.7 Routed Traffic

If you are running a dual homed firewall (in this context a non-routing firewall), you should already be preventing the firewall from routing traffic. This can be done by opening the **Control Panel** and double-clicking on **Network** to open network configuration of Windows NT. Then, select **Protocols** and double-click on **TCP/IP Protocol** to open the TCP/IP configuration. Choose the **Routing** tab and check off the **Enable IP Forwarding** as shown below.

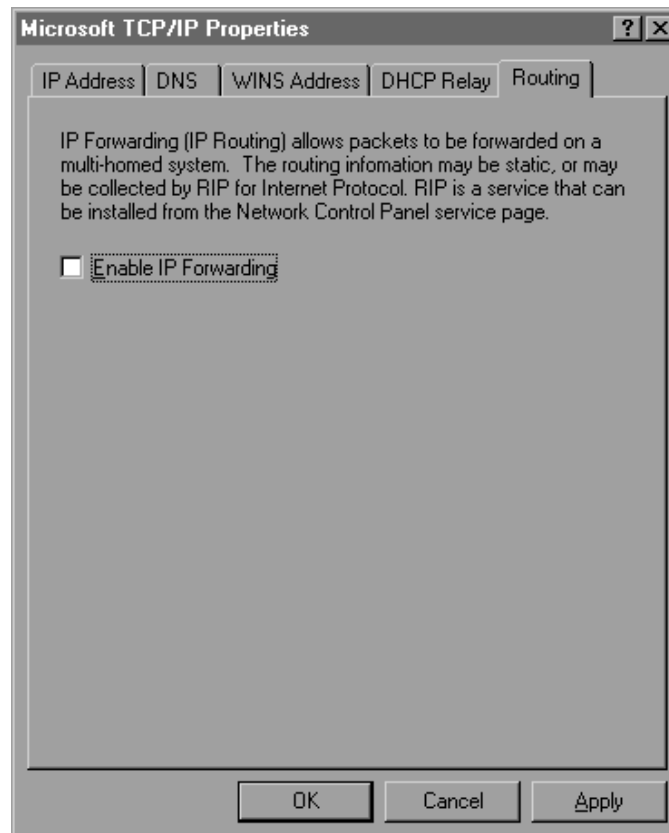


Figure 145. Disable IP Forwarding on Windows NT

Click on **OK** to save the changes. The new configuration will be activated after you reboot the firewall.

However, it is a good idea to have a rule to block routed traffic as well, in case the IP forwarding option is activated by mistake. The following connection will also prevent routing.

Connection: Block Routed Traffic

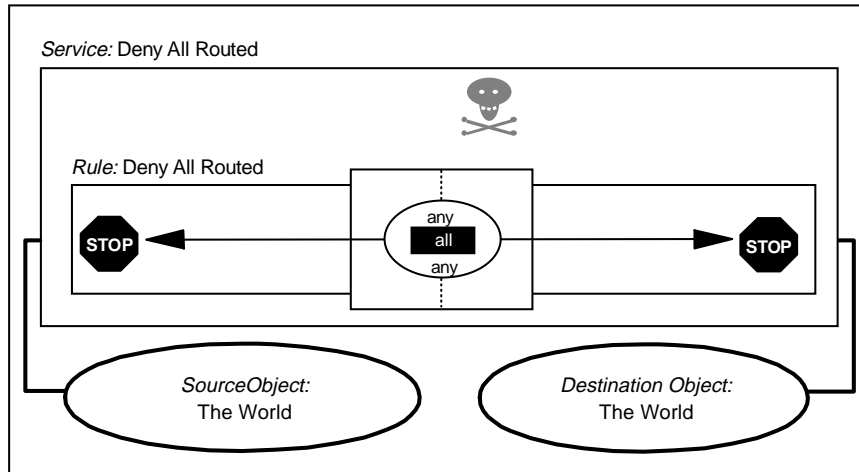


Figure 146. Prevent Routed Traffic

The filter definition resulting from this connection is:

```
deny 0 0 0 0 all any 0 any 0 both routed both
```

9.3 Telnet

Telnet is a protocol used to emulate terminal sessions. Telnet servers normally use TCP port 23, while the client uses one of the non-privileged ports (starting from port 1024). Telnet uses passwords in order to authenticate the user. These passwords cross the network unencrypted. The Telnet client may also be used to access other TCP-based services, for example, electronic mail (SMTP).

We recommend you do not allow Telnet from the non-secure network to the secure network. As Telnet sends the password unencrypted, an outsider can use a sniffer to grab passwords from the network and use them later. In the past there have been attacks to the main Internet nodes, in which attackers have installed sniffers in order to capture passwords (see CERT Advisory CA-95:18).

Even if you use one-time passwords, there are some serious security concerns. There have been attacks in which intruders anticipate a user connection being made. Once the user is authenticated the intruder hijacks the connection and starts to send its own packets (see CERT Advisory CA-95:01).

There are three possible scenarios that will be covered in this section:

1. Telnet from firewall to secure network
2. Telnet using Proxy
3. Telnet using SOCKS

9.3.1 Telnet From Firewall to Secure Network

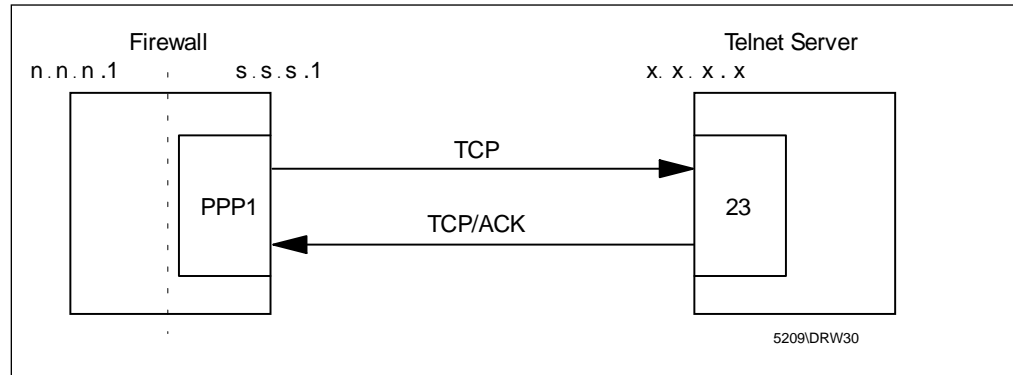


Figure 147. Telnet from Firewall to Secure Network

The following connection permits you to log in to a host in the secure network from the firewall, using Telnet. The first rule in the service allows traffic to be initiated by the firewall to any Telnet server in the secure network (TCP port 23). The second rule allows the Telnet Server (TCP port 23) to reply. The connection uses standard, predefined rules and service.

Connection: Telnet from Firewall to SN

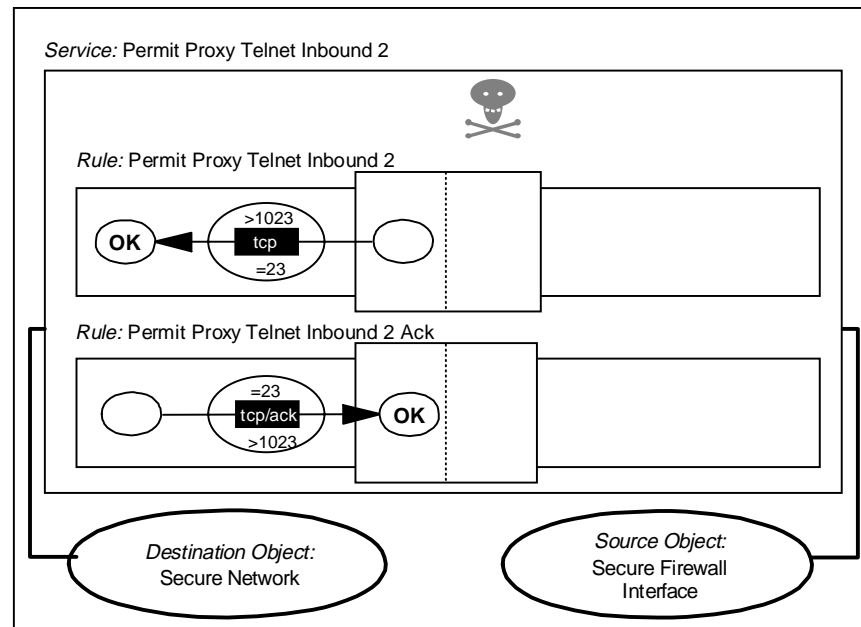


Figure 148. Connection Definitions for Telnet to Secure Network

This connection generates the following filter rules:

```

permit s.s.s.1 0xffffffff s.s.s.s sm.sm.sm.sm tcp gt 1023 eq 23 secure local outbound
permit s.s.s.1 sm.sm.sm.sm s.s.s.1 0xffffffff tcp/ack eq 23 gt 1023 secure local inbound

```

Notice that we are using the tcp/ack format to prevent misuse of port 23 to establish a session (for example, a connection from port 23 to the SOCKS server in port 1080).

9.3.2 Telnet Using Proxy

Telnet with a proxy is a two-step connection. First the user logs into the firewall with a normal Telnet session. Once on the firewall, they use Telnet again to reach the final destination (which might be some other TCP service, not necessarily Telnet on port tcp/23). The second log in may be entered explicitly or automatically. We will describe how to configure the Telnet and FTP proxy server in Chapter 13, "Configuring Proxy Services and SOCKS" on page 293.

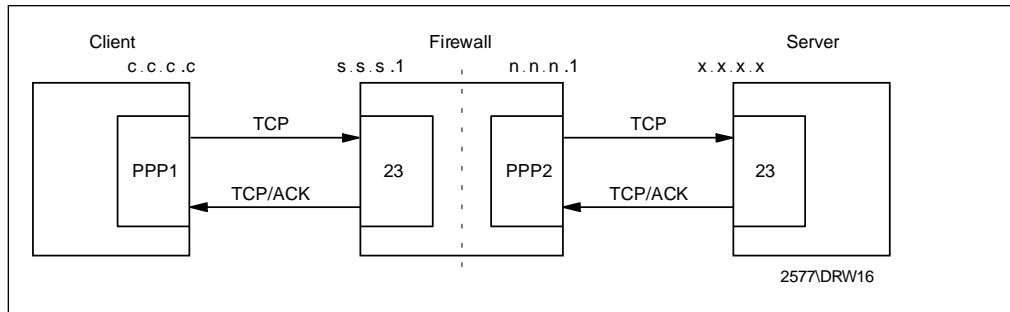


Figure 149. Telnet from Secure Network to Non-secure Network Using Proxy

A connection definition can only reference one pair of network objects (source and destination). Because there are two parts to the session, one on either side of the firewall, we need to define two connections to describe it as follows:

Connection: Telnet from Secure Network to Firewall

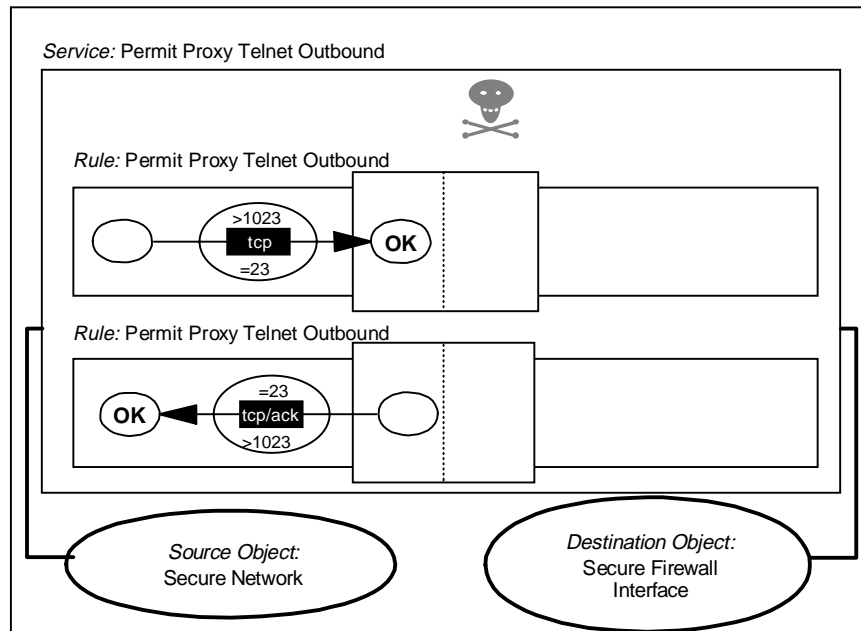


Figure 150. Connection from Secure Network to Proxy Telnet Server

Connection: Telnet from Firewall to NSN

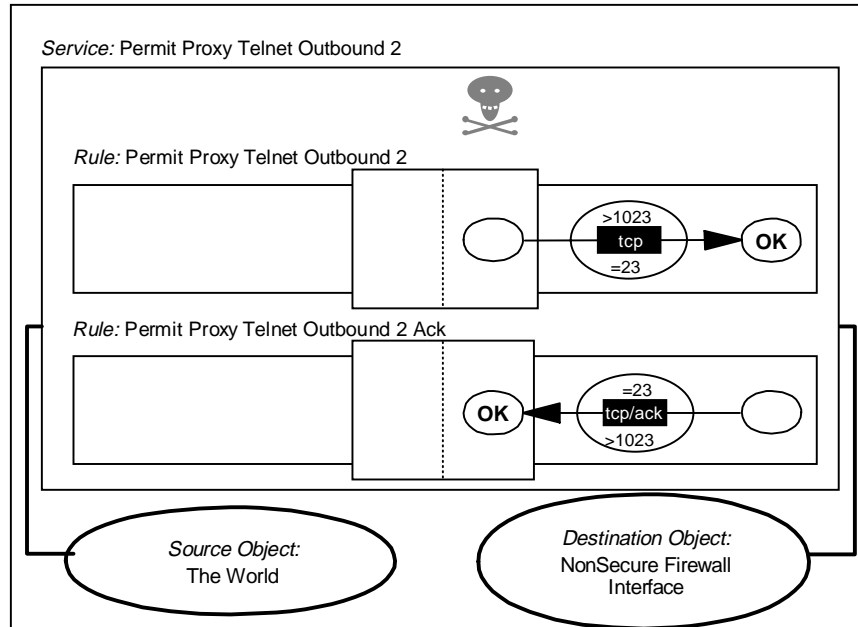


Figure 151. Connection from Proxy to Remote Telnet Server

These connections generate the following filter rules:

```
# Telnet from secure network to the Firewall
permit s.s.s.s sm.sm.sm.sm s.s.s.1 0xffffffff tcp    gt 1023 eq 23 secure local inbound
permit s.s.s.1 0xffffffff s.s.s.s sm.sm.sm.sm tcp/ack eq 23 gt 1023 secure local outbound

# Telnet from Firewall to the Non-secure Network
permit n.n.n.1 0xffffffff 0 0 tcp    gt 1023 eq 23 nonsecure local outbound
permit 0 0 n.n.n.1 0xffffffff tcp/ack eq 23 gt 1023 nonsecure local inbound
```

9.3.3 Telnet Using SOCKS

Telnet using SOCKS is a two-step connection:

1. The user client connects to the SOCKS server on the firewall (using TCP port 1080).
2. The firewall establishes a connection to the final destination.

The difference is the firewall does not ask for the user ID and password for user authentication. So, SOCKS is transparent for the users. And the switching is done in the transport layer which is different from what PROXY does in the application layer.

You will find a more detailed description of this process in 13.7.2, "Configuring SOCKS Services" on page 326.

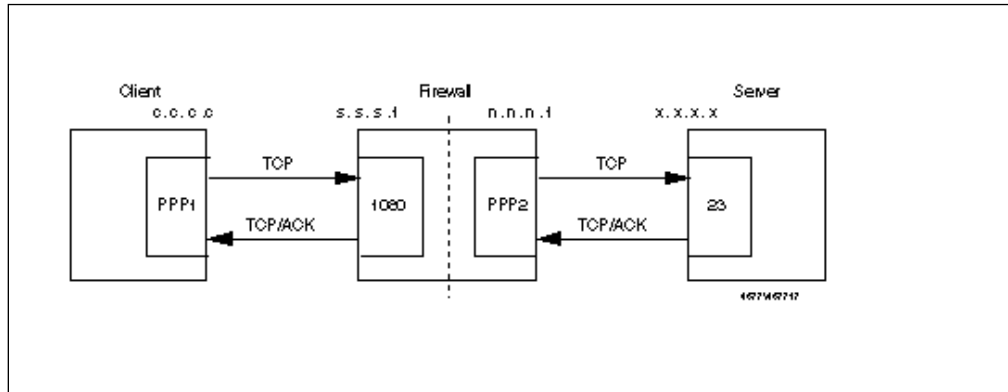


Figure 152. Telnet from Secure Network to Non-secure Network Using SOCKS

The first part of the session is very similar to the proxy case, except using the SOCKS port tcp/1080 instead of tcp/23, as follows:

Connection: Socks from Secure to Firewall

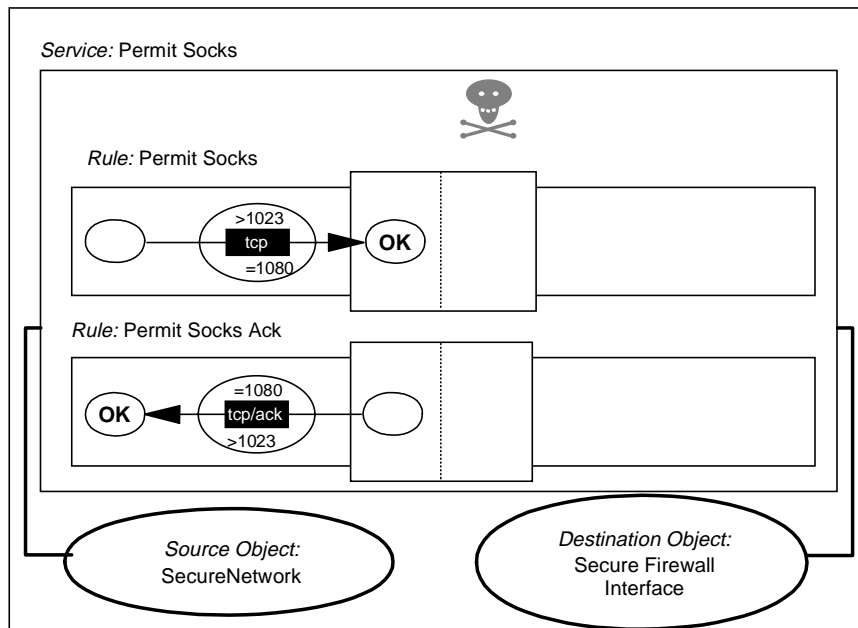


Figure 153. Connection from Secure Network to SOCKS Server

The second connection in the session is the same as Figure 151 on page 173.

These connections generate the following filter rules:

```
# Connection from the client to the SOCKS Server
permit s.s.s.s sm.sm.sm.sm s.s.s.s 0xffffffff tcp gt 1023 eq 1080 secure local inbound
permit s.s.s.s 0xffffffff s.s.s.s sm.sm.sm.sm tcp/ack eq 1080 gt 1023 secure local outbound

# Telnet from Firewall to the Non-secure Network
permit n.n.n.n 0xffffffff 0 0 tcp gt 1023 eq 23 nonsecure local outbound
permit 0 0 n.n.n.n 0xffffffff tcp/ack eq 23 gt 1023 nonsecure local inbound
```


9.4 FTP: File Transfer Protocol

FTP also uses stream-oriented (TCP) sessions. However, it is more complicated than Telnet since it actually uses two different ports, one for the commands and the other for the data. It also has two different possibilities for establishing the connection, called normal-mode FTP (also called active-mode) and passive-mode FTP. Both FTP modes are explained in the next two sections.

9.4.1 Normal Mode

The server is listening on tcp/21. The client, using a nonprivileged port, connects to the server establishing the control session. When the user enters a command such as `dir`, `get` or `put`, the server, using port 20 (ftp-data), establishes a connection to a nonprivileged port of the client.

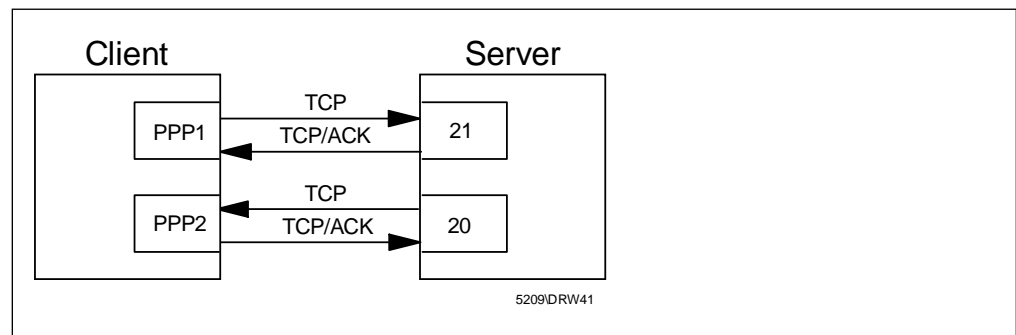


Figure 154. FTP Normal Mode

In this case there is an incoming connection that must be allowed from port 20 to an unknown port. This allows outsiders to misuse port 20, so we recommend that you allow incoming connections only to the proxy server where you can limit the number of services that you provide.

9.4.2 Passive Mode

The server is listening on tcp/21. The client, using a nonprivileged port, connects to the server establishing the control session. When the user enters a command such as `dir`, `get` or `put`, the client establishes a second connection from a nonprivileged port to a nonprivileged port of the server.

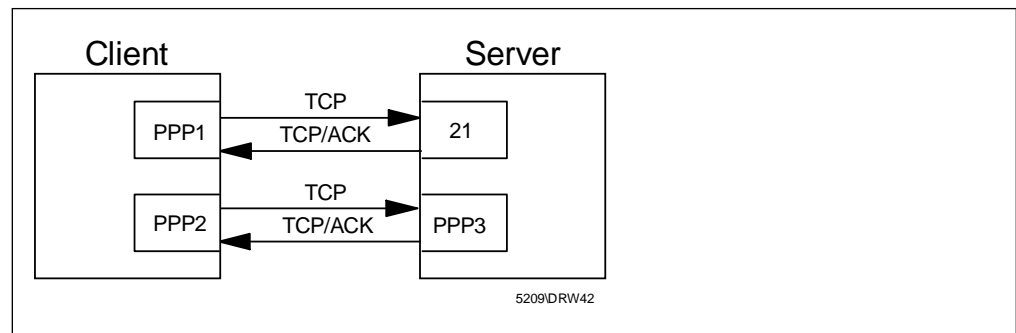


Figure 155. FTP Passive Mode

This avoids the problem of the incoming connection, but requires you to access nonfixed ports for the data channel connection. See RFC 1579, *Firewall Friendly FTP* for a complete discussion about FTP in a firewall context.

9.4.3 FTP from Secure Network to Non-secure Network Using Proxy

This can be allowed, but remember to protect any TCP services that the firewall is providing on non-privileged ports. Normally this is only SOCKS, unless you have added some other service.

In this case the FTP client connects to an FTP server on the firewall. Once there, it is authenticated by the server in the normal way. Once the user uses the quote site command, the proxy connects to the server. To this point, we only have control sessions from the client to port 21 on the firewall and from the non-secure side of the firewall to port 21 on the target server. When the user gets or puts a file, the FTP client will specify a mode for the transfer (either normal or passive). The FTP proxy will use the same type of transfer to connect to the final server.

The following is an example of this with normal-mode FTP.

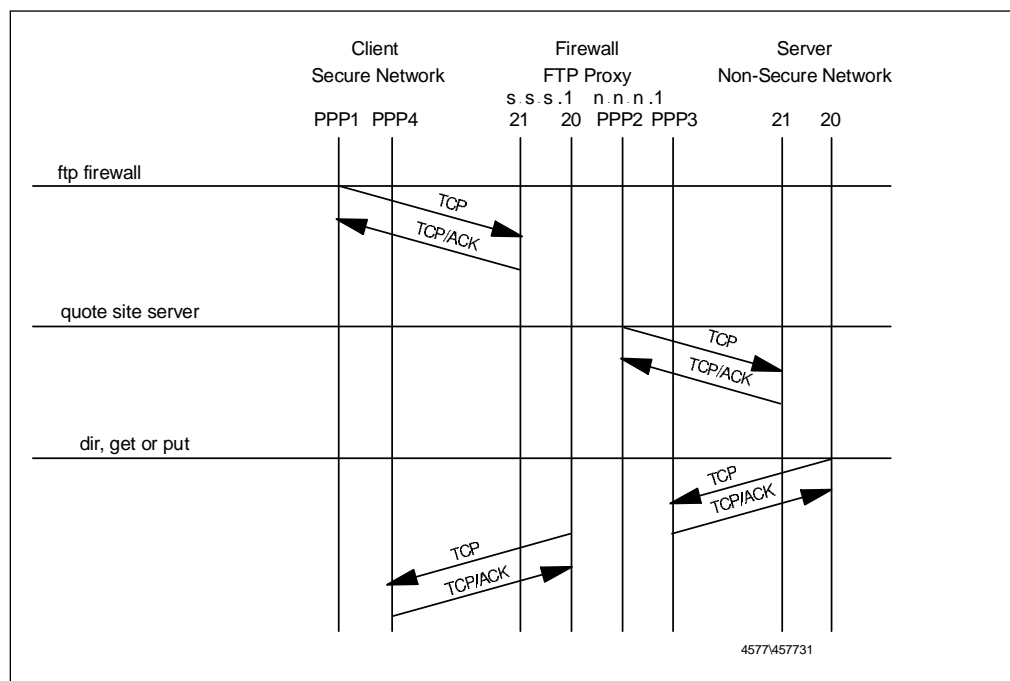
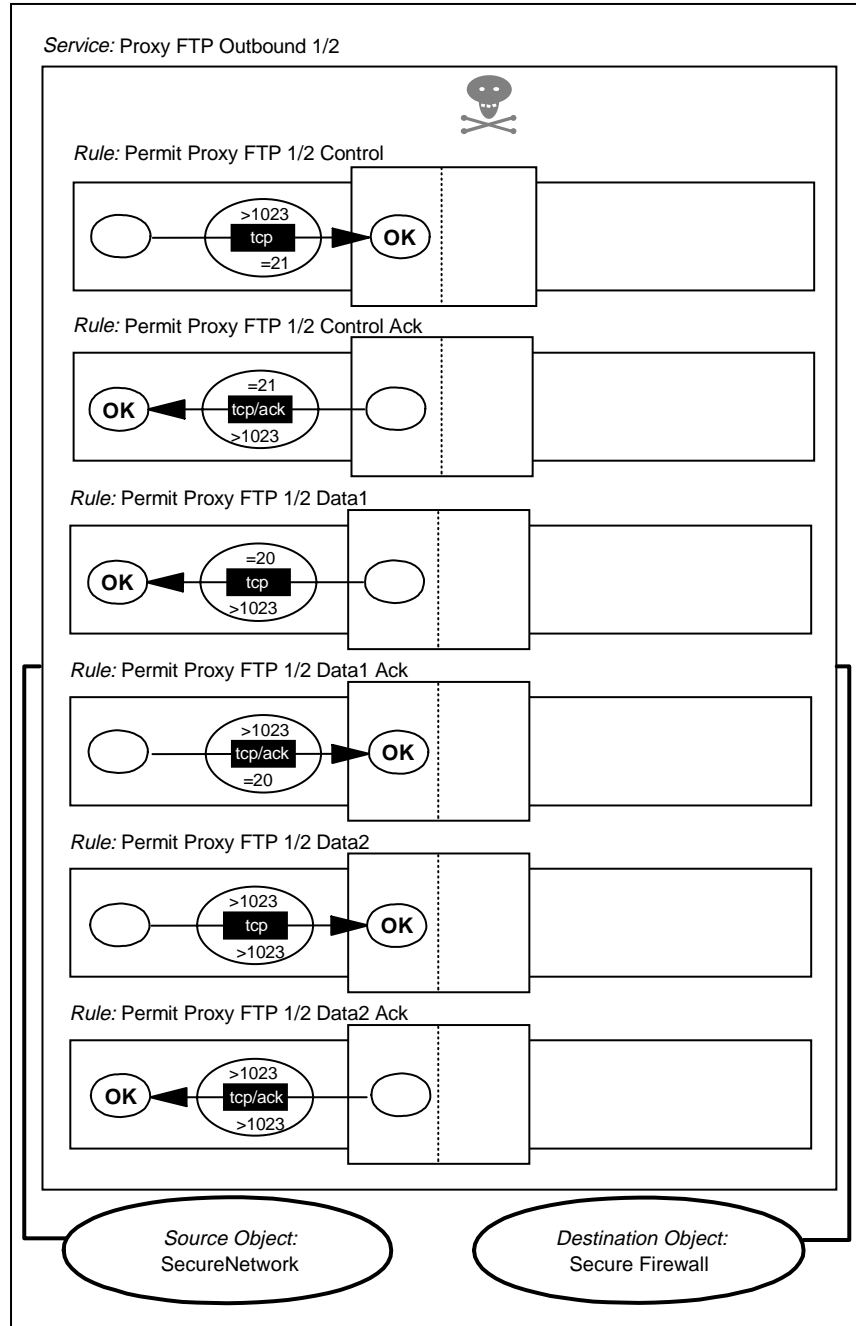


Figure 156. Normal Mode FTP from Secure Network to Non-secure Network Using Proxy

The following connections provide FTP access from the secure network to the non-secure network using the proxy server:

Connection: Proxy FTP from SN to Firewall



2577\SNGPIX17

Figure 157. First Connection, Secure Network to FTP Proxy

Connection: Proxy FTP from Firewall to NSN

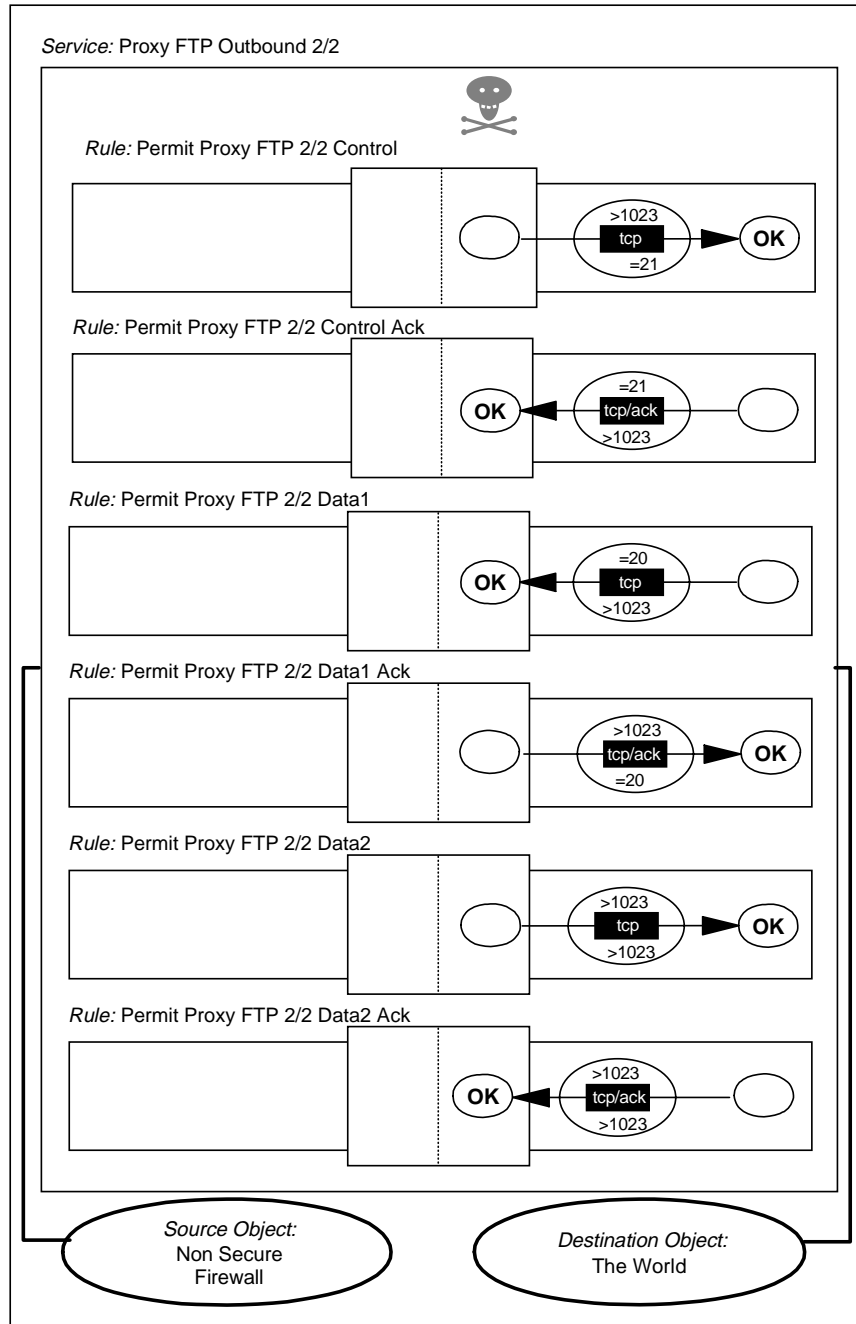


Figure 158. Second Connection, FTP Proxy to Non-secure Network

The resulting filter rules are as follows:

```
# FTP From SN to NSN. FTP Control Session from Client to Firewall (1)
permit s.s.s.s sm.sm.sm.sm s.s.s.1 0xffffffff tcp gt 1023 eq 21 secure local inbound
permit s.s.s.1 0xffffffff s.s.s.s sm.sm.sm.sm tcp/ack eq 21 gt 1023 secure local outbound

# FTP From SN to NSN. FTP Control Session from Firewall to Server (2)
permit n.n.n.1 0xffffffff 0 0 tcp gt 1023 eq 21 nonsecure local outbound
permit 0 0 n.n.n.1 0xffffffff tcp/ack eq 21 gt 1023 nonsecure local inbound

# Normal Mode
```

```

# FTP From SN to NSN. FTP Data Session from Server to Firewall (3)
permit 0 0 n.n.n.1 0xffffffff tcp eq 20 gt 1023 nonsecure local inbound
permit n.n.n.1 0xffffffff 0 0 tcp/ack gt 1023 eq 20 nonsecure local outbound

# FTP From SN to NSN. FTP Data Session from Firewall to Client (4)
permit s.s.s.1 0xffffffff s.s.s.s sm.sm.sm.sm tcp eq 20 gt 1023 secure local outbound
permit s.s.s.s sm.sm.sm.sm s.s.s.1 0xffffffff tcp/ack gt 1023 eq 20 secure local inbound

# Passive Mode

# FTP From SN to NSN. FTP Data Session from Server to Firewall (3)
permit n.n.n.1 0xffffffff 0 0 tcp gt 1023 gt 1023 nonsecure local outbound
permit 0 0 n.n.n.1 0xffffffff tcp/ack gt 1023 gt 1023 nonsecure local inbound

# FTP From SN to NSN. FTP Data Session from Firewall to Client (4)
permit s.s.s.1 0xffffffff s.s.s.s sm.sm.sm.sm tcp gt 1023 gt 1023 secure local inbound
permit s.s.s.s sm.sm.sm.sm s.s.s.1 0xffffffff tcp/ack gt 1023 gt 1023 secure local outbound

```

9.4.4 FTP from Secure Network to Non-secure Network Using SOCKS

The rules for FTP using SOCKS are very similar to the ones for the proxy. The difference is that in this case, the client only uses passive mode. You have to change the destination of the client's connection from 21 and gt 1023 to the SOCKS server on port 1080 and define only the passive mode rules.

You can create this connection by using the standard SOCKS connection definition for your secure network and a connection which uses a service that can be created easily by copying the service *Permit Proxy FTP Outbound 2/2* and removing two rules (see Figure 159 on page 180).

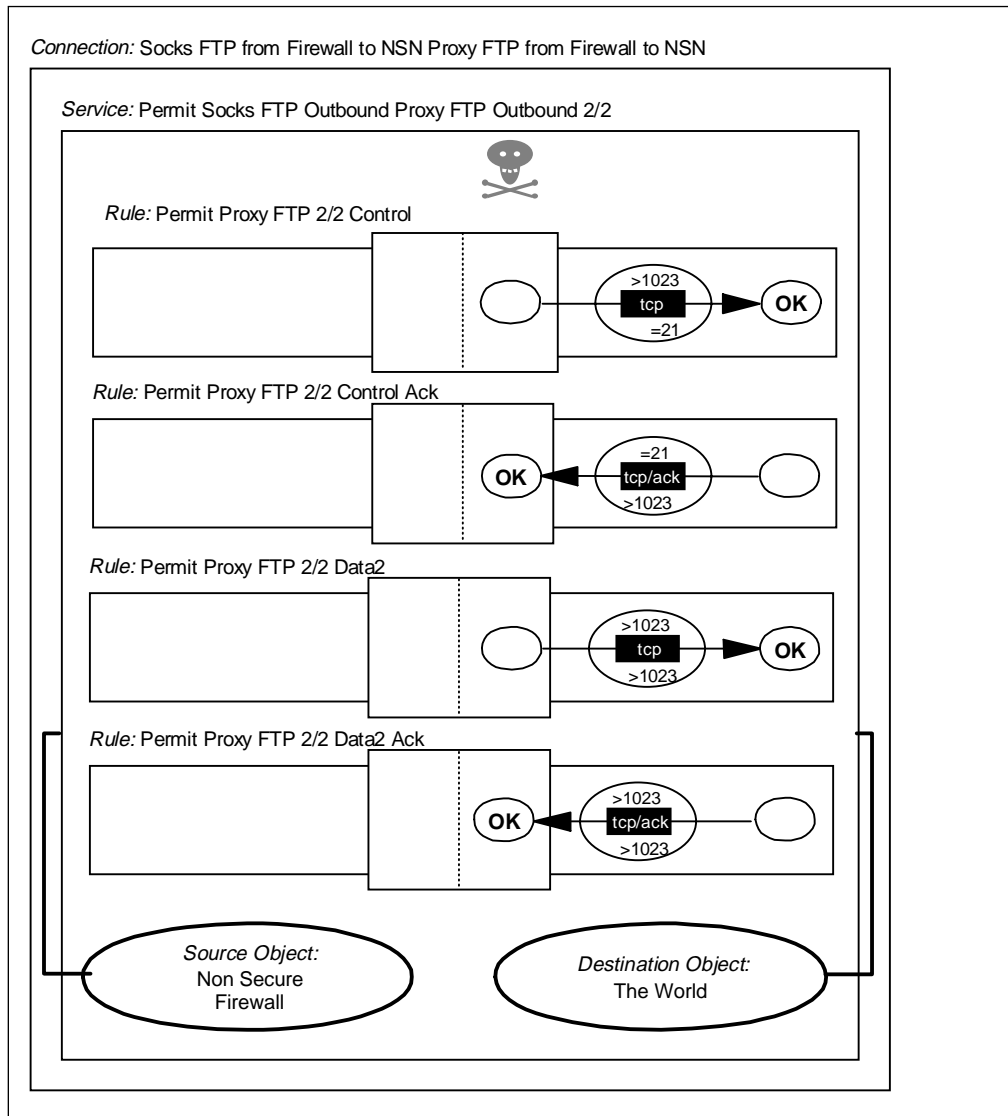


Figure 159. FTP from Secure Network To Non-secure Network Using SOCKS

These connections generate the following filter rules:

```
# FTP From SN to NSN. FTP Control & Data Session from Client to SOCKS Server.
permit s.s.s.s sm.sm.sm.s.s.s.1 0xffffffff tcp gt 1023 eq 1080 secure local inbound
permit s.s.s.s.1 0xffffffff s.s.s.s sm.sm.sm.sm tcp/ack eq 1080 gt 1023 secure local outbound
```

```
# FTP From SN to NSN. FTP Control Session from Firewall to FTP Server
permit n.n.n.1 0xffffffff 0 0 tcp gt 1023 eq 21 nonsecure local outbound
permit 0 0 n.n.n.1 0xffffffff tcp/ack eq 21 gt 1023 nonsecure local inbound
```

```
# FTP From SN to NSN. FTP Data Session from Firewall to FTP Server (Passive Mode)
permit n.n.n.1 0xffffffff 0 0 tcp gt 1023 gt 1023 nonsecure local outbound
permit 0 0 n.n.n.1 0xffffffff tcp/ack gt 1023 gt 1023 nonsecure local inbound
```

9.4.5 Normal Mode FTP from Firewall to Secure Network

The following connection allows the outbound control session to port 21 and the inbound data session from port 20. All of the rules within the connection are from the predefined set.

Connection: FTP from Firewall to SN

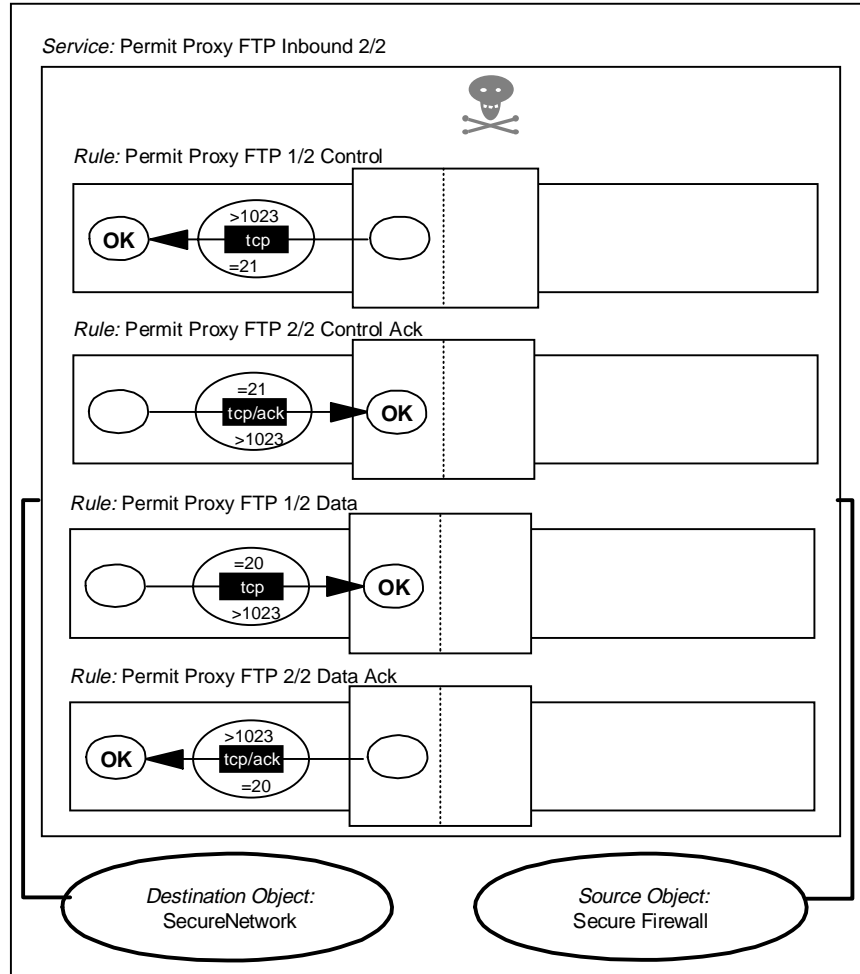


Figure 160. FTP from Firewall to Secure Network

These connections generate the following filter rules:

```
# FTP From Firewall to SN. FTP Control Session
permit s.s.s.1 0xffffffff s.s.s.s sm.sm.sm.sm tcp gt 1023 eq 21 secure local outbound
permit s.s.s.s sm.sm.sm.sm s.s.s.1 0xffffffff tcp/ack eq 21 gt 1023 secure local inbound

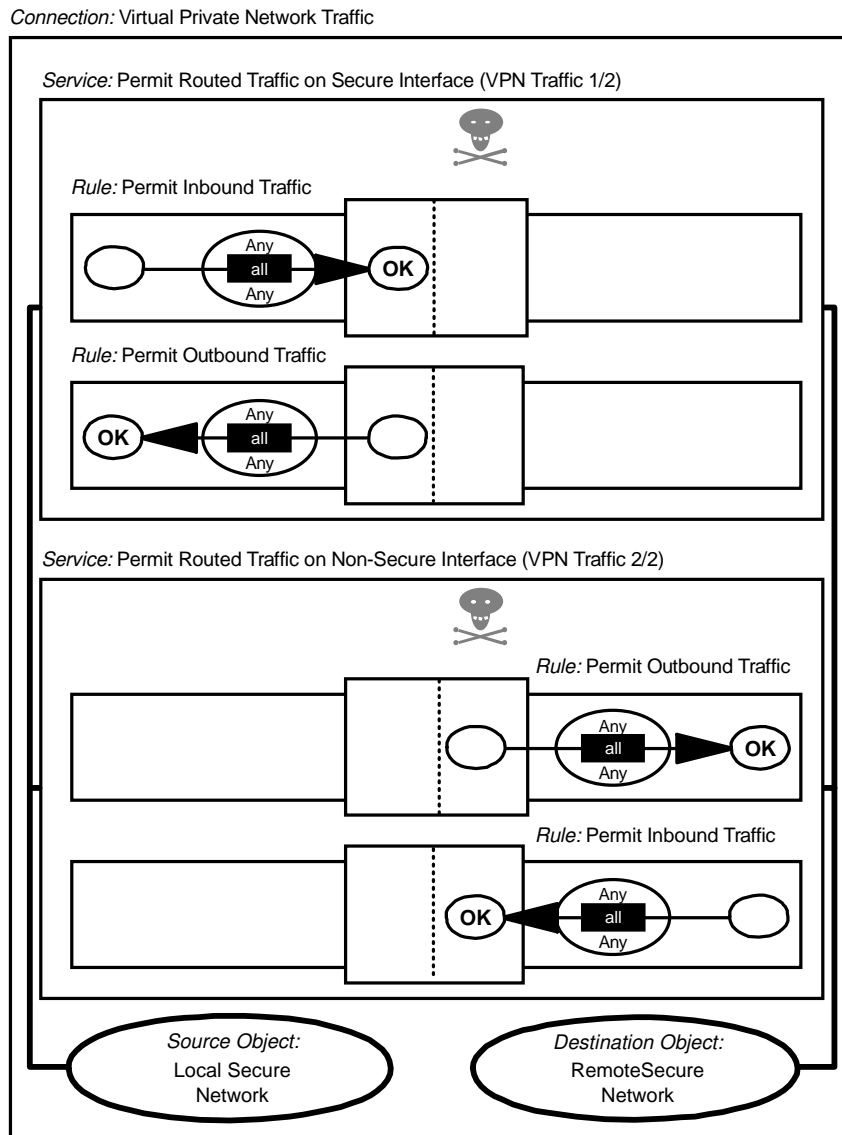
# FTP From Firewall to SN. FTP Data Session
permit s.s.s.s sm.sm.sm.sm s.s.s.1 0xffffffff tcp eq 20 gt 1023 secure local inbound
permit s.s.s.1 0xffffffff s.s.s.s sm.sm.sm.sm tcp/ack gt 1023 eq 20 secure local outbound
```

9.5 VPN: Virtual Private Network

A Virtual Private Network is an extension of the private network, it allows secure communication between two secure networks over a nonsecure network. To allow this communication and make it secure we use authentication and/or encryption. To establish a tunnel we need to configure two connections, one for the data traffic between the secure clients and one for the encapsulation between the two tunnel endpoints, to protect the contents of the packets. Each packet passes two times through the filter mechanism, once in the clear and once in its encapsulated form to be transported through the tunnel, therefore it requires a different connection to permit this passage.

9.5.1 Data Traffic Connection

The connection for the data traffic is between the two secure networks or secure clients in those networks. One client in the secure network will send packets to another secure client in the remote secure network. These packets will be routed through the firewall into the defined tunnel.



5209\PRSS096

Figure 161. VPN Data Traffic

Destination Object is the network or host to be connected to at the remote end of the VPN tunnel.

These connections generate the following filter rules:

```
# VPN Traffic from Secure Network to Firewall (VPN Traffic 1/2)
permit s.s.s.s sm.sm.sm.sm r.r.r.r.rm.rm.rm.rm all any 0 any 0 secure route inbound
permit r.r.r.r.rm.rm.rm.rm s.s.s.s sm.sm.sm.sm all any 0 any 0 secure route outbound
```

```
# VPN Traffic from the Firewall into the Tunnel (VPN Traffic 2/2)
permit r.r.r.r.rm.rm.rm.rm s.s.s.s sm.sm.sm.sm all any 0 any 0 non-secure route outbound t=ti
permit s.s.s.s sm.sm.sm.sm r.r.r.r.rm.rm.rm.rm all any 0 any 0 non-secure route inbound t=ti
```


The first service, VPN Traffic 1/2, permits all the traffic from the secure network to the secure interface of the firewall.

The packets are routed to the nonsecure interface of the firewall where the second service, VPN Traffic 2/2, will put them into the defined tunnel. At the end of the rule we see that the tunnel id parameter is added, t=ti, to specify which tunnel will be used.

9.5.2 Encapsulation Connection

The encapsulation connection is between the two tunnel endpoints. A new IP header and an AH or ESP header (or both) will be added to the original IP header.

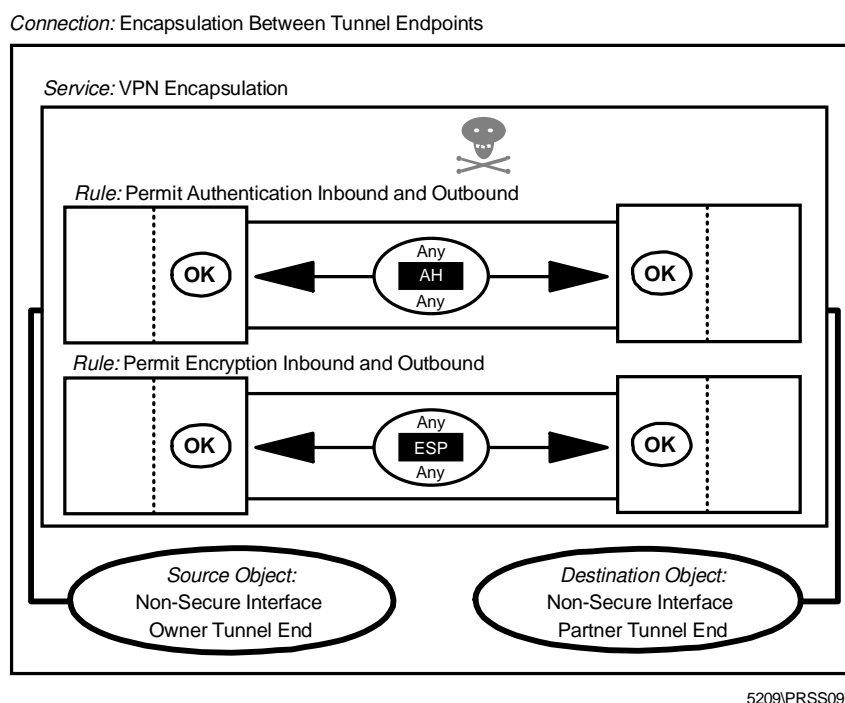


Figure 162. VPN Encapsulation

For the encapsulation connection we have four new rules, two for the authentication and two for the encryption.

```
# VPN Encapsulated Traffic between the Two Tunnel Endpoints
permit t.t.t.1 0xffffffff t.t.t.2 0xffffffff ah any 0 any 0 non-secure local both
permit t.t.t.2 0xffffffff t.t.t.1 0xffffffff ah any 0 any 0 non-secure local both
permit t.t.t.1 0xffffffff t.t.t.2 0xffffffff esp any 0 any 0 non-secure local both
permit t.t.t.2 0xffffffff t.t.t.1 0xffffffff esp any 0 any 0 non-secure local both
```

The first two rules permit authentication in both directions between the tunnel endpoints for in and outbound packets and the last two do the same for the encryption.

9.6 SMTP: Simple Mail Transfer Protocol

Mail formatted using the Simple Mail Transfer Protocol (SMTP) is one of the largest contributors to Internet traffic. In TCP/IP terms, SMTP is a straightforward, stream-oriented application. The SMTP server (the receiver of incoming mail) listens for connections on TCP port 25. The client uses any TCP port, usually one of the nonreserved ports (above 1023).

As the SMTP protocol doesn't provide any control, a very important point in this service is to educate your users. Any user on the Internet could send e-mail messages with a forged origin, so you should tell your users that the apparent source of an e-mail message cannot always be trusted, unless the message is signed by a strong authentication mechanism such as Pretty Good Privacy (PGP). Fortunately, IBM eNetwork Firewall for Windows NT will log the IP host name of the message sender, not the one that is specified in the SMTP connection handshake (in the HELO command)

There are many possible configurations for handling mail using IBM eNetwork Firewall for Windows NT. We will discuss a few of them in Chapter 12, "Mail Handling" on page 269. For the purpose of this filter rule example, we'll only discuss the rules that will be activated by Security Policy and the rules that we recommend you use.

The connections shown in Figure 163 are the standard rules from IBM eNetwork Firewall for Windows NT. These rules will be activated when we enable Secure Mail Proxy traffic from Security Policy.

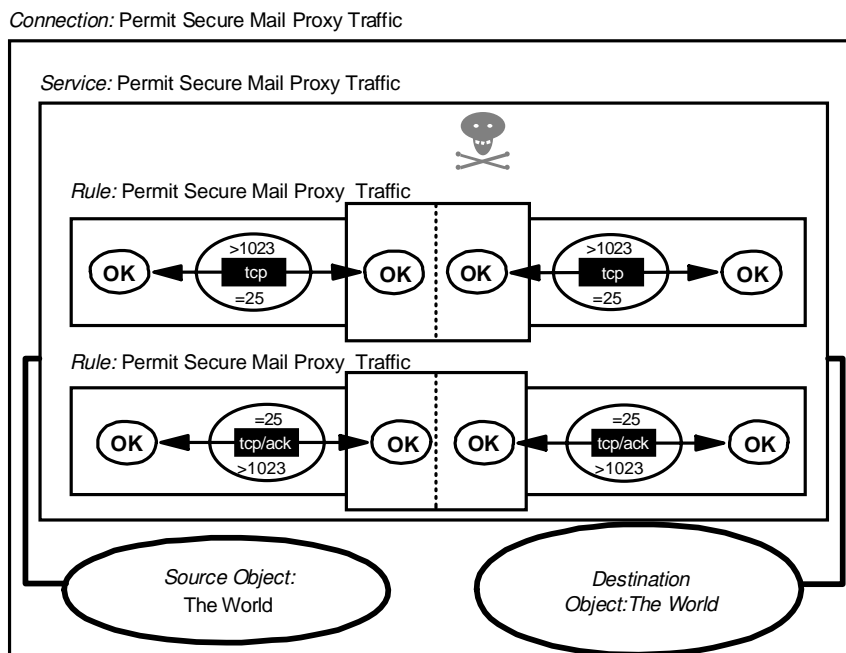


Figure 163. Default Connections of Secure Mail Proxy

These connections generate the following filter rules:

```
# (SECURITY POLICY) Permit Secure Mail Proxy traffic through Firewall
permit 0 0 0 0 tcp gt 1023 eq 25 both local both
permit 0 0 0 0 tcp/ack eq 25 gt 1023 both local both
```

As you can see from the above, the default rules are not safe at all. These rules allow all hosts in the secure network to connect to port tcp/25. To avoid unexpected connections to the firewall, we recommend you use an SMTP gateway in the internal network (M.M.M.M). The configuration will be changed as shown in Figure 164 on page 185.

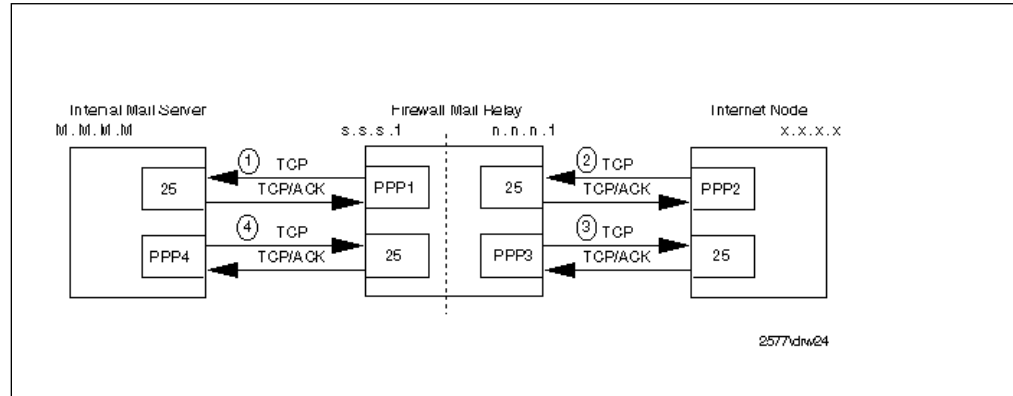


Figure 164. SMTP Mail Configuration

Since the sessions between the mail daemons may be set up in either direction, depending on where the mail arrives from, the connections have been designed to cater to either case.

Connection: SMTP Internal

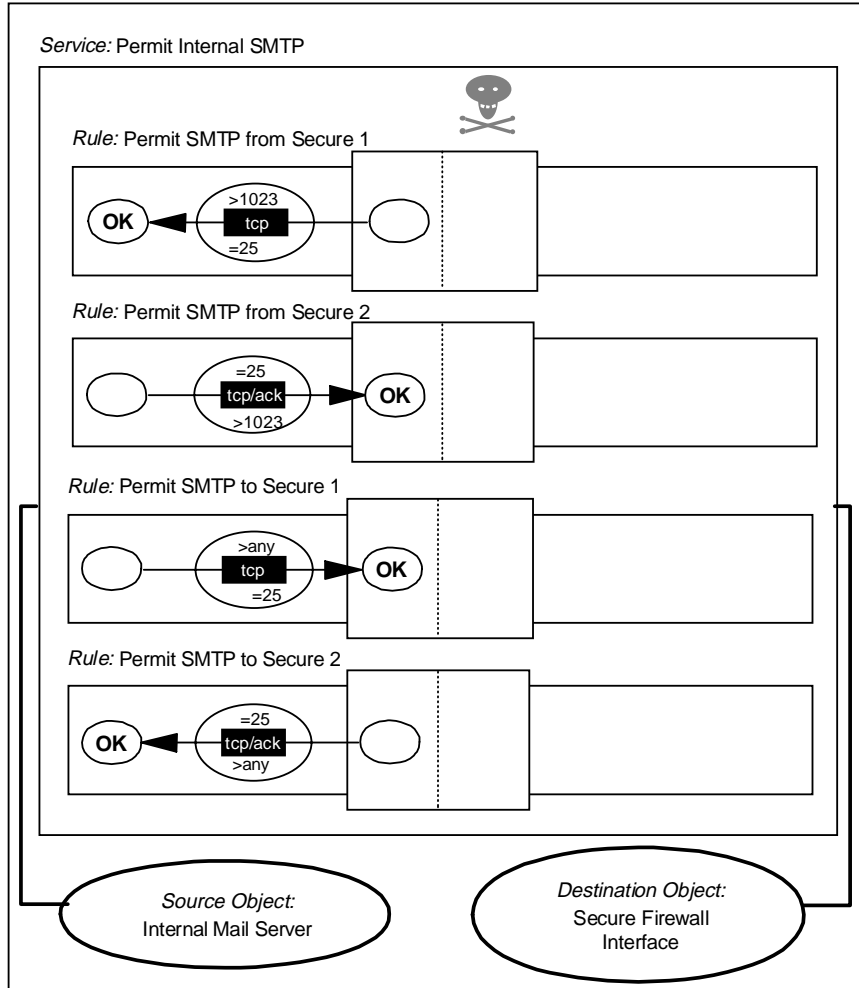


Figure 165. SMTP Mail from Secure Mail Server to Firewall SafeMail Gateway

Connection: SMTP External

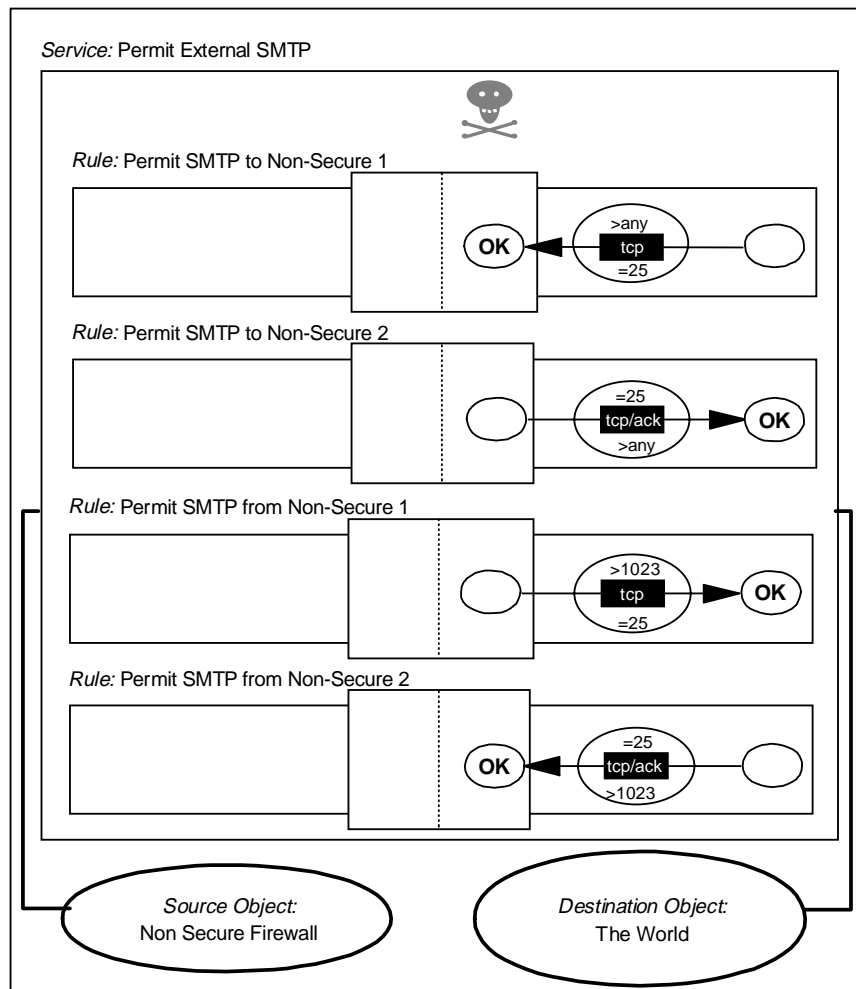


Figure 166. SMTP Mail from Firewall SafeMail Gateway to Non-secure Network

The resulting filter rules are as follows:

```
# Mail from the Firewall to the Internal Mail Server (1)
permit s.s.s.1 0xffffffff M.M.M.M 0xffffffff tcp gt 1023 eq 25 secure local outbound
permit M.M.M.M 0xffffffff s.s.s.1 0xffffffff tcp/ack eq 25 gt 1023 secure local inbound

# Mail from the NSN to the Firewall (2)
permit 0 0 n.n.n.1 0xffffffff tcp any 0 eq 25 nonsecure local inbound
permit n.n.n.1 0xffffffff 0 0 tcp/ack eq 25 any 0 nonsecure local outbound

# Mail from the Firewall to the NSN (3)
permit n.n.n.1 0xffffffff 0 0 tcp gt 1023 eq 25 nonsecure local outbound
permit 0 0 n.n.n.1 0xffffffff tcp/ack eq 25 gt 1023 nonsecure local inbound

# Mail from the Internal Mail Server to the Firewall (4)
permit M.M.M.M 0xffffffff s.s.s.1 0xffffffff tcp any 0 eq 25 secure local inbound
permit s.s.s.1 0xffffffff M.M.M.M 0xffffffff tcp/ack eq 25 any 0 secure local outbound
```

9.7 DNS: Domain Name Server

For reasons we have already discussed (see Chapter 11, “Domain Name Service (DNS)” on page 247), it is not a good idea to allow non-secure nodes to have unrestricted access to the name-to-address mapping of the nodes in the secure network. However, you will want some of your machines to be visible, and you will want your own name server to be able to access external name servers. We will further discuss the various DNS configurations in Chapter 11, “Domain Name Service (DNS)” on page 247.

Like SMTP, IBM eNetwork Firewall for Windows NT through Security Policy provides a single check box to enable DNS queries. The connections are shown in Figure 167.

Connection: Permit DNS Queries

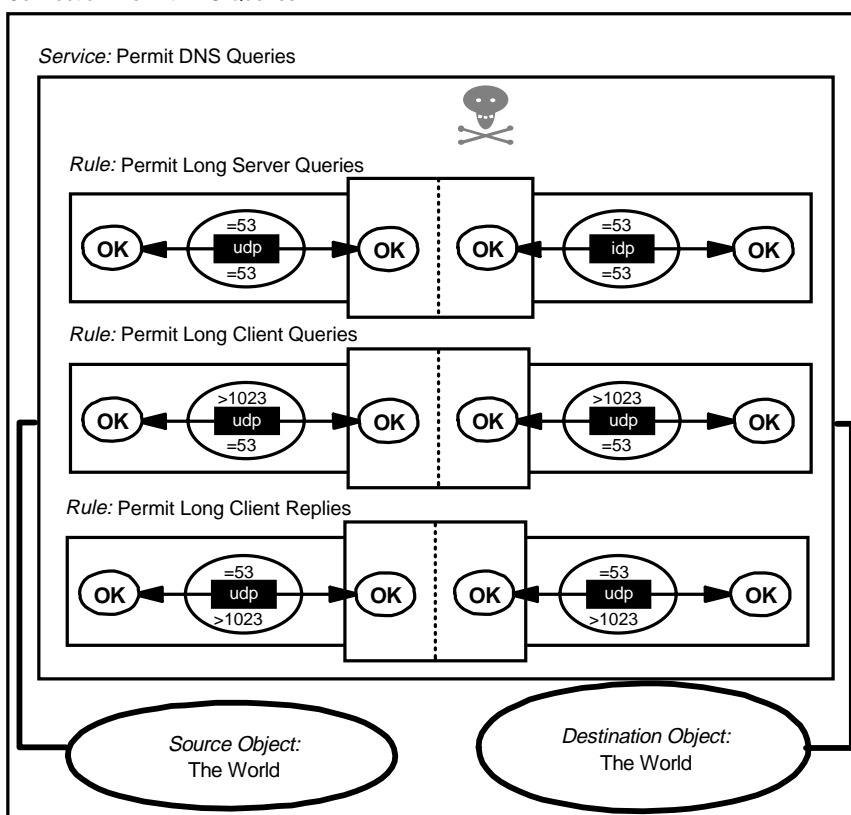


Figure 167. Default Connections of DNS Queries

And the resulting filter rules are as follows:

```
# Default Rules of DNS Queries
permit 0 0 0 0 udp eq 53 eq 53 both local both
permit 0 0 0 0 udp gt 1023 eq 53 both local both
permit 0 0 0 0 udp eq 53 gt 1023 both local both
```

You should also consider the DNS zone transfers and DNS long queries that use the TCP protocol instead of UDP. See 11.4.3, “Configuration of DNS in Firewall” on page 263 for more details on this subject.

9.8 NNTP: Network News Transfer Protocol

NNTP is a protocol used to transfer news. It is a TCP service in which the server listens on port 119. If your users employ an NNTP-capable reader with SOCKS support, you can use the configuration shown in Figure 168.

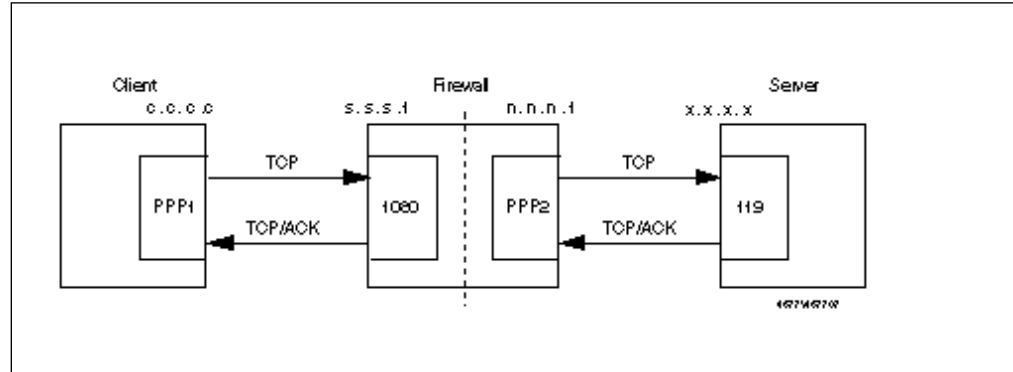


Figure 168. NNTP Client Using SOCKS Server

This requires two connection definitions. The first is a standard SOCKS client connection, as shown in Figure 160 on page 181. The second is shown below. Note that the rules used in this connection are not part of the predefined set, so they have to be manually added.

Connection: NNTP from Firewall to Anywhere

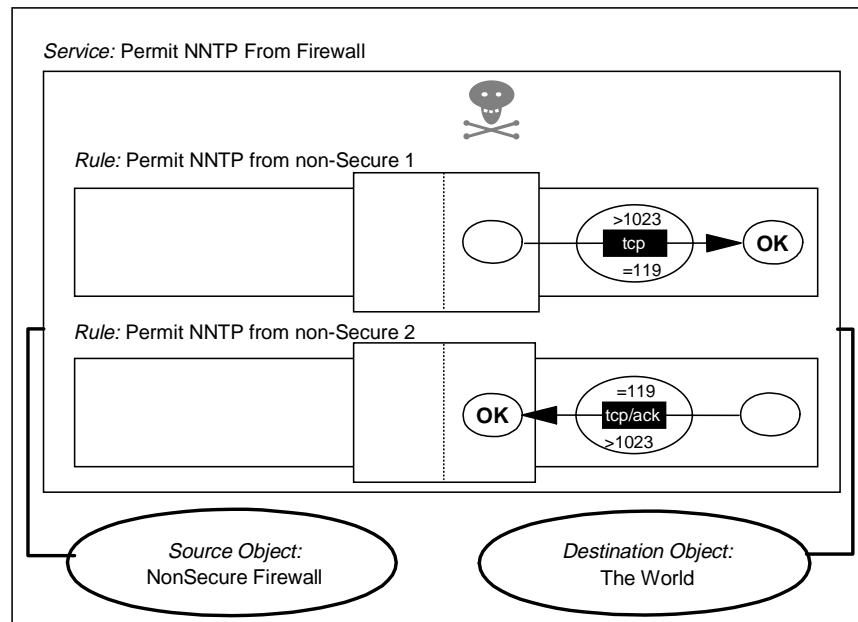


Figure 169. Connection to a Non-secure NNTP Server

The filter rules generated by this configuration are as follows:

```
# Connection from the client to the SOCKS Server
permit s.s.s.s sm.sm.sm.sm s.s.s.1 0xffffffff tcp    gt 1023 eq 1080 secure local inbound
permit s.s.s.1 0xffffffff s.s.s.s sm.sm.sm.sm tcp/ack eq 1080 gt 1023 secure local outbound

# Connection from Firewall to the News Server in the Non-secure Network
permit n.n.n.1 0xffffffff 0 0 tcp    gt 1023 eq 119  nonsecure local outbound
permit 0 0 n.n.n.1 0xffffffff tcp/ack eq 119  gt 1023 nonsecure local inbound
```

In the above case, the news server is outside the firewall, either in a remote network or in the DMZ. Often it is more convenient to have a news server in the secure network, in order to store news locally. This avoids duplicated traffic and allows you to restrict the news groups to only those that you would like to provide. You may also want to provide some internal news groups, which are not broadcast outside the the firewall. If you do put a news server inside the secure network, you will receive news from an external news feeder to the news server. There is no standard proxy for NNTP in IBM eNetwork Firewall for Windows NT, so the simplest approach is to allow IP forwarding of these packets.

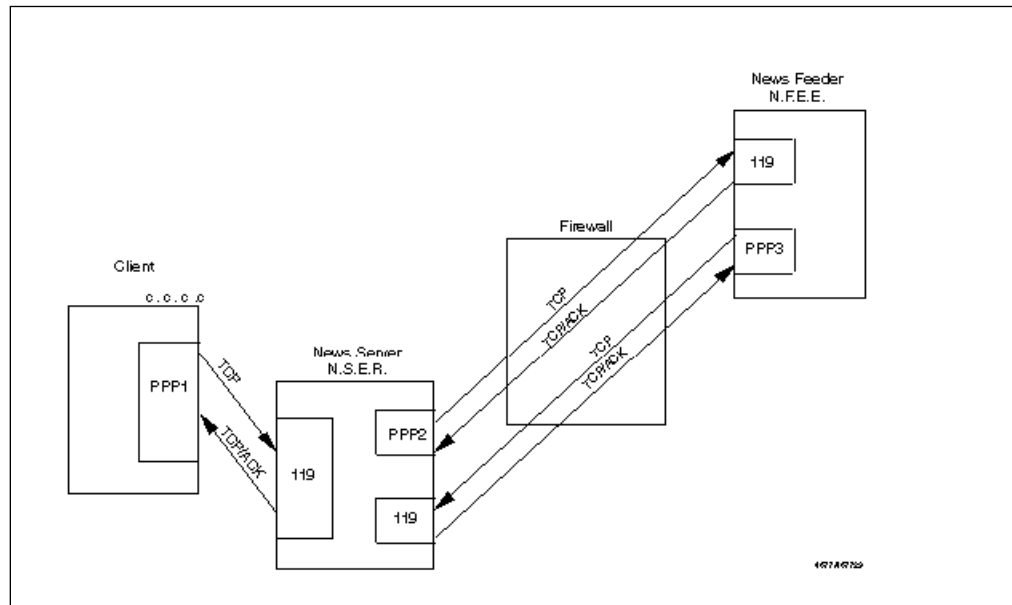


Figure 170. NNTP: News Server with IP Forwarding Enabled

The following connection definition can be used. As with the previous NNTP case, the rule definitions must be manually added, but in this case we can exploit the facility to reverse the direction of a rule, so in fact the first and third rules are identical, as are the second and fourth.

Connection: Routed NNTP

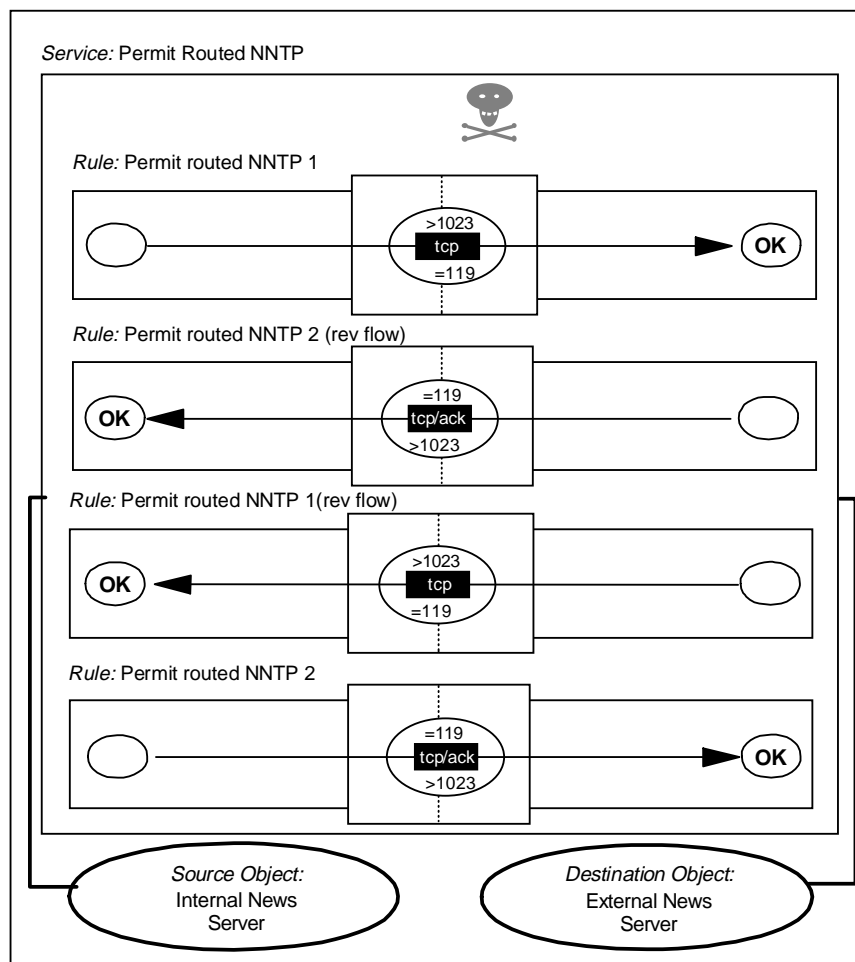


Figure 171. NNTP Routed through the Firewall

The following filter rules are implemented by this connection:

```
# Outgoing Postings. From Internal News Server to News Feeder.
permit N.S.E.R 0xffffffff N.F.E.E 0xffffffff tcp gt 1023 eq 119 secure route inbound
permit N.F.E.E 0xffffffff N.S.E.R 0xffffffff tcp/ack eq 119 gt 1023 secure route outbound
permit N.S.E.R 0xffffffff N.F.E.E 0xffffffff tcp gt 1023 eq 119 nonsecure route outbound
permit N.F.E.E 0xffffffff N.S.E.R 0xffffffff tcp/ack eq 119 gt 1023 nonsecure route inbound

# Incoming News. From News Feeder to Internal News Server.
permit N.F.E.E 0xffffffff N.S.E.R 0xffffffff tcp gt 1023 eq 119 secure route inbound
permit N.S.E.R 0xffffffff N.F.E.E 0xffffffff tcp/ack eq 119 gt 1023 secure route outbound
permit N.F.E.E 0xffffffff N.S.E.R 0xffffffff tcp gt 1023 eq 119 nonsecure route outbound
permit N.S.E.R 0xffffffff N.F.E.E 0xffffffff tcp/ack eq 119 gt 1023 nonsecure route inbound
```

If you would prefer to run a dual-homed firewall with no IP forwarding being permitted at all, then you need to either have a SOCKSified news server or a proxy in the firewall that communicates with both the news feeder and the news server and acts as a relay between the two. There is a sample program that does the later; you should read *Protect and Survive Using the IBM Firewall for AIX 3.1*, SG24-2577 for more details on this program. The source code is available. We did not test this solution.

9.9 HTTP - World Wide Web Sessions

The World Wide Web (WWW) is a collection of sites and services, loosely connected by a network of inter-document links. The normal way to access this network is using a graphical browser interface such as Netscape Navigator, Microsoft Internet Explorer or NCSA Mosaic.

From a technical point of view the main vehicle for WWW documents is the *HyperText Transfer Protocol* (HTTP). This is a lightweight protocol for requesting and receiving information using any encoding mechanism recognized by both the client and server. HTTP is a stateless protocol; that is, the server retains no continuing session information about a client. This has the benefit of allowing great simplicity and extensibility at the expense of some network bandwidth overhead.

The basic HTTP exchange is a simple request-response sequence:

1. A *request* is sent from an unprivileged port on the client to the server on TCP port 80.

The request may contain either a simple GET action, or a more complex *method* description (used, for example, when the client is sending a form that the user has filled in). The request also contains information about the client, such as the browser software in use and a list of the document formats that it can handle.

2. The *response* packets flow on the reverse path to the request (that is, from port 80 back to the requesting client port).

The response is usually a document encoded using *Multi-Purpose Internet Mail Extensions* (MIME). The document will be written in one of the formats acceptable to the browser, most commonly the *HyperText Markup Language* (HTML), a document composition language which allows you to imbed links to other documents and to other graphical and multimedia objects.

WWW hyperlinks do not always lead to other HTTP connections. Often they will take you to an anonymous FTP or gopher site. When you select the hyperlink to such a site, the WWW browser software invokes the appropriate service under the covers. So when you plan to provide this service, you should also consider the other related protocols.

As you can see from the previous description, most of the complexity of WWW lies in the higher-layer application code (which is a source for security concern, particularly as the content of Web documents becomes more interactive and as Web browsers become smarter). See *Safe Surfing: How to Build a Secure World Wide Web Connection*, SG24-4564, for more information.

9.9.1 Possible Scenarios

HTTP is a relatively new protocol. It was developed when firewalls were commonly in use on the Internet, so usually the Web browsers are proxy and SOCKS aware. As always, we aim to break all sessions at the firewall boundary, so a SOCKS or proxy arrangement is best.

There is another important advantage in the use of proxy servers. Some proxy servers also cache Web documents and save the most frequently accessed ones. This is a considerable performance improvement for the users and also helps to reduce network traffic (and hence also costs, when your Internet connection

charges are traffic-based). IBM eNetwork Firewall for Windows NT includes a Web proxy server, but this is *not* a caching version. The reason for this is that placing a complex application server of this kind on the firewall would break the KISS principle.

If you are going to provide a Web server for access by the rest of the world, it should be outside your secure network, ideally in a DMZ. The reason for this is that such servers are, by the nature of the application, likely to be complex and therefore vulnerable to attack. If you can isolate your server by putting it outside your secure network, you are limiting the damage that such break-ins can do. Many examples of attacks on Web servers have been demonstrated, so you should follow the CERT advisories to check if you are exposed. The possible scenarios that are covered in this section are:

1. SOCKSified client
2. Internal HTTP Proxy Server using SOCKS
3. HTTP Proxy Server in the firewall
4. Internal HTTP Proxy Server using IP forwarding
5. External HTTP Proxy Server

9.9.2 SOCKSified Client

In this case, clients in the secure network connect to the SOCKS server on the firewall, which relays their sessions into the Internet. One disadvantage of this configuration is that there is no caching of documents, so if multiple users load the same document, it will be retrieved multiple times.

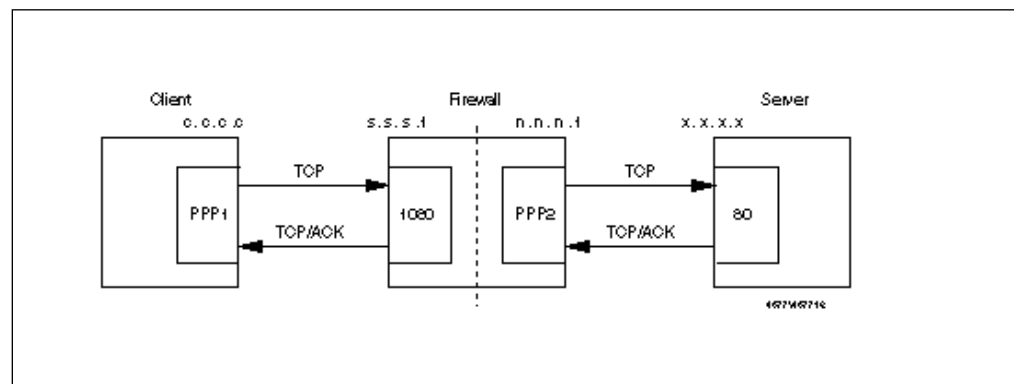


Figure 172. HTTP from Secure Network Using a SOCKSified Client

This configuration requires two connections. The first is the familiar SOCKS client connection shown in Figure 153 on page 174. The second connection is as follows:

Connection: HTTP from Firewall to Anywhere

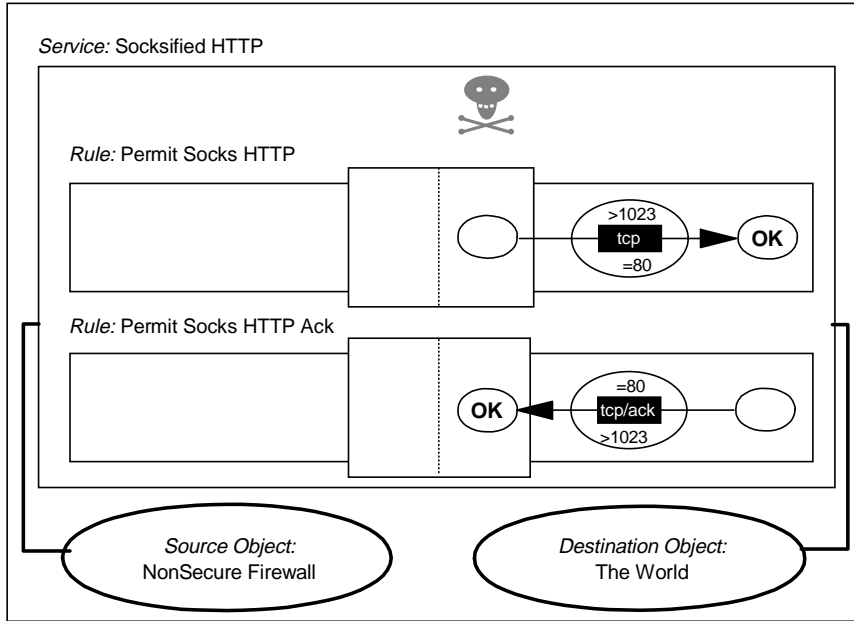


Figure 173. HTTP from SOCKS to Non-secure Server

The equivalent filter definitions are as follows:

```
# Connection from the client to the SOCKS Server
permit s.s.s.s sm.sm.sm.sm s.s.s.1 0xffffffff tcp gt 1023 eq 1080 secure local inbound
permit s.s.s.1 0xffffffff s.s.s.s sm.sm.sm.sm tcp/ack eq 1080 gt 1023 secure local outbound

# Connection from Firewall to the Server in the Non-secure Network
permit n.n.n.1 0xffffffff 0 0 tcp gt 1023 eq 80 nonsecure local outbound
permit 0 0 n.n.n.1 0xffffffff tcp/ack eq 80 gt 1023 nonsecure local inbound
```

9.9.3 Internal HTTP Proxy Server Using SOCKS

This extends the SOCKS configuration by placing a proxy server in the secure network. Clients connect to the proxy, which serves documents from cache if it is available or establishes a connection across the SOCKS relay, if not. The proxy server has to have SOCKS support built in for this configuration to work correctly. Many proxy servers provide this facility, including the IBM Internet Connection family.

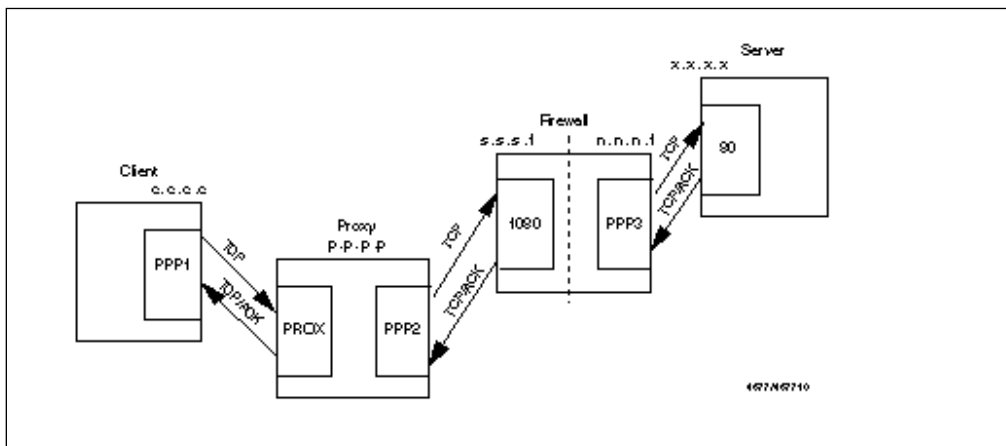


Figure 174. HTTP from Secure Network Using a SOCKSified HTTP Proxy

The filtering rules for this case are the same as for the SOCKSified Client, except that instead of allowing connections from every client in the secure network, you can restrict them to the proxy server only. You could also restrict access to the SOCKS server so that *only* the proxy server can use it for HTTP. To do this, you would put the following lines in the sockd.conf file:

```
# Restrict SOCKSified HTTP Connections (port 80) only to HTTP Proxy
permit p.p.p.p 255.255.255.255 eq 80
deny 0.0.0.0 0.0.0.0 eq 80
```

9.9.4 HTTP Proxy Server in the Firewall

The two previous examples (9.9.2, "SOCKSified Client" on page 193 and 9.9.3, "Internal HTTP Proxy Server Using SOCKS" on page 194) both use the SOCKS server to relay sessions through the firewall. In each case you could replace it with the HTTP proxy server that comes as part of the IBM eNetwork Firewall for Windows NT by simply using rules that permit access to the proxy port (usually tcp/8080) in place of the SOCKS port (tcp/1080).

9.9.5 Internal HTTP Proxy Server Using IP Forwarding

This has the disadvantage that you must enable IP forwarding in the firewall, but it is relatively secure because the filter rules are very restrictive (just one host and outbound connection only).

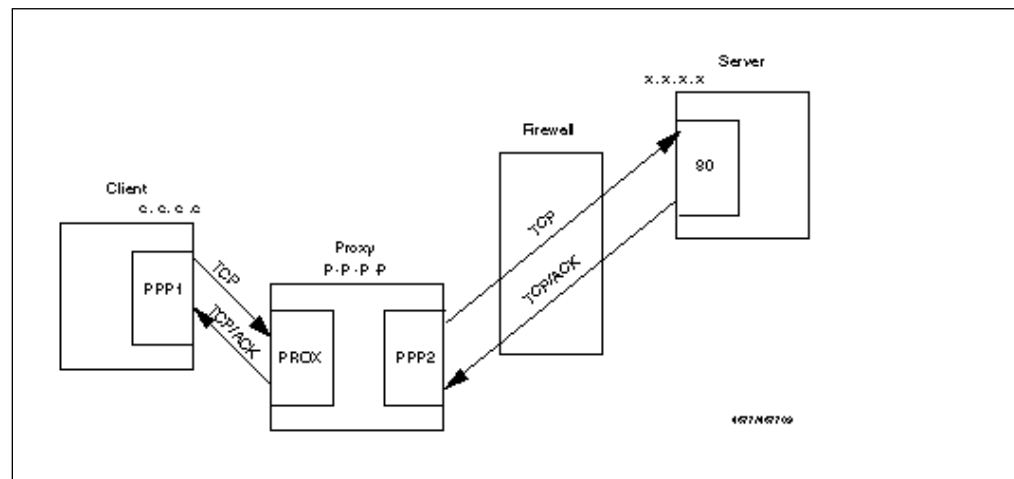


Figure 175. HTTP from Secure Network: HTTP Proxy with IP Forwarding Enabled

9.9.6 External HTTP Proxy Server

This case is also a good solution, especially if you are using a DMZ configuration. You can put the proxy server in the DMZ for caching. Ideally you should only allow your users to access the proxy using SOCKS.

The clients request a document using SOCKS from the proxy that is outside the secure network. The proxy does the caching, and you don't have to allow routing in the firewall, so you can run a dual-homed configuration.

In this case, you will need your Web browsers to support concurrent SOCKS and HTTP proxy. You can accomplish this if your client TCP/IP implementation has built-in SOCKS support (for example, in Warp V4) by specifying a proxy in the

browser configuration menu, and then using the SOCKS run time from the TCP/IP configuration options (that is, not the browser SOCKS option).

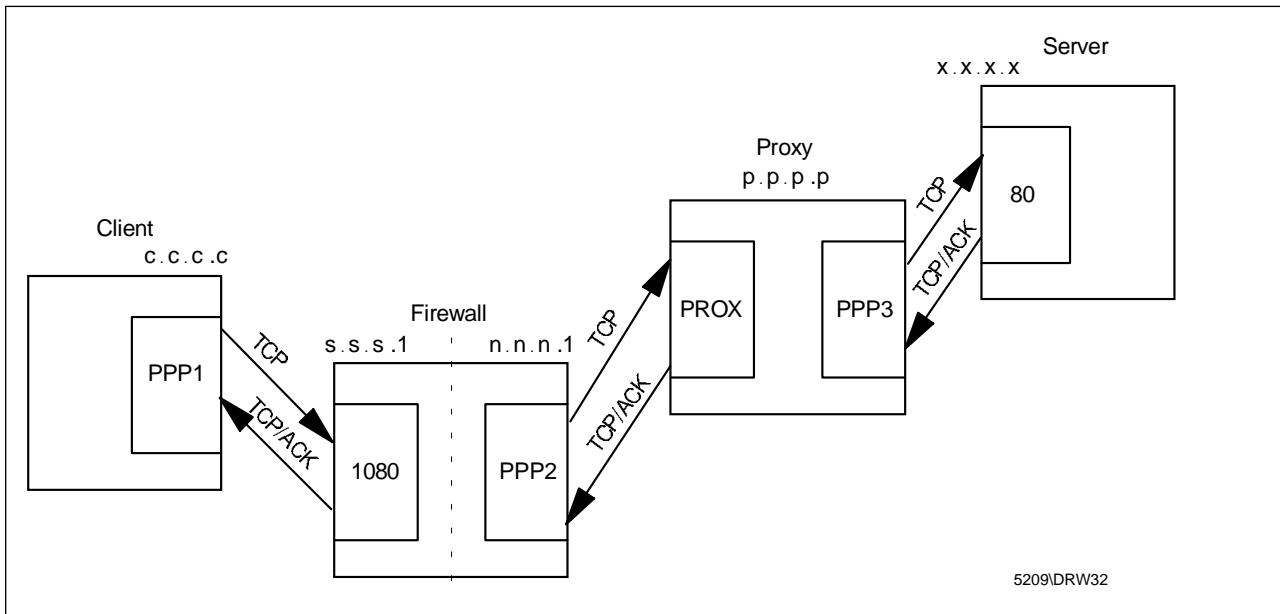


Figure 176. HTTP Proxy in the Non-secure Network (Preferably Using DMZ)

In this kind of configuration it is very important to treat the proxy server in the DMZ as part of your firewall. It should be monitored closely for any sign of attack. If a cracker were to manage to take control of it he or she could use it to set up a *Web Spoof*. This means that the attacker has control over all communications to anywhere in the Web. He or she can choose to simply act as a relay, recording passwords and information as it flows past, or could intervene and insert his own messages or malicious executables into the pages that your users retrieve.

9.10 SSL: Secure Sockets Layer

SSL is a protocol developed by Netscape Communications Corporation along with RSA Data Security, Inc. It attempts to provide an alternative for the standard TCP/IP socket API to resolve problems of authentication and encryption. Although this means that, in theory, it could be used to protect any TCP/IP application, it is used almost exclusively for HTTP. It is used in Web-based applications to transport confidential data such as credit card numbers and is also able to provide authentication of the Web server, using public key certificates. Client authentication is defined in the SSL standard, but it is not commonly used yet.

From a firewall perspective SSL is a standard TCP service in which the client uses any nonprivileged port and the server uses port 443. Follow the recommendations for HTTP and substitute port 443 in place of 80.

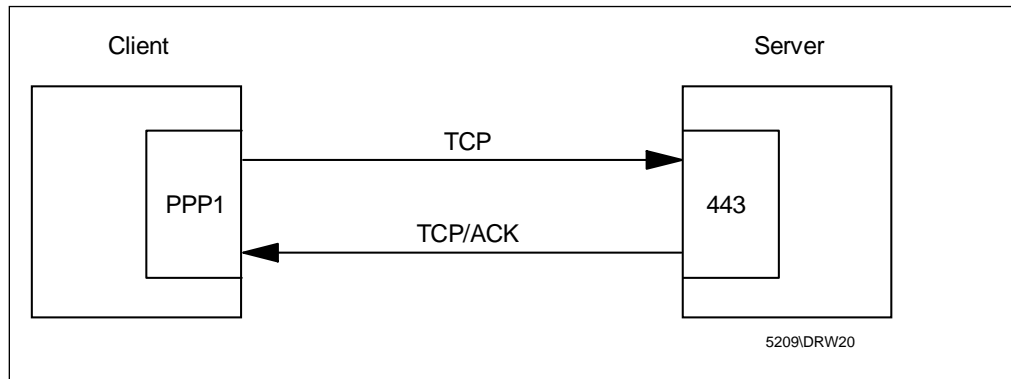


Figure 177. SSL Connection between Client and Server

9.11 S-HTTP

S-HTTP is a protocol designed in order to correct the lack of authentication and encryption of HTTP. S-HTTP is very rarely used in practice. SSL is a much easier protocol to administer and, although it is less flexible than S-HTTP, it has proved adequate for most secure Web applications.

From a filtering point of view, S-HTTP is identical to HTTP. The server listens for connections on tcp port 80, and the clients use any nonprivileged port. The server will realize if it is an HTTP connection or an S-HTTP connection from the URL: if it starts with http, it is HTTP, if it starts with https, it is S-HTTP. Web documents retrieved with S-HTTP should not be cached in a proxy server, because the proxy server cannot determine what S-HTTP options are appropriate.

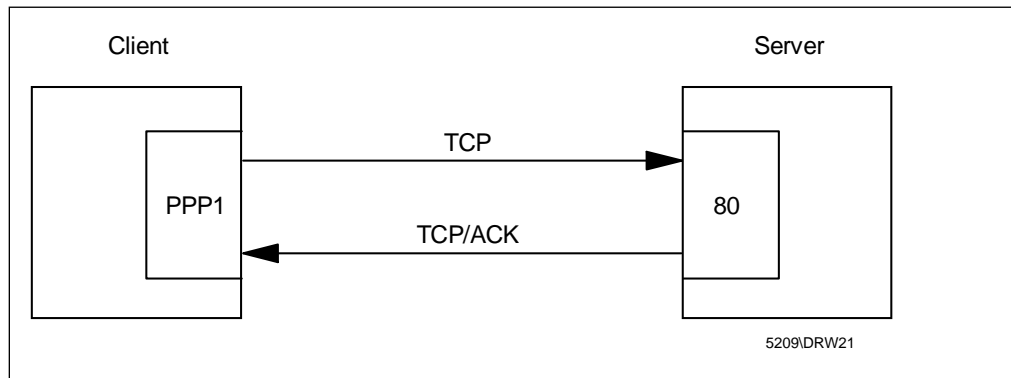


Figure 178. Secure HTTP Connection between Client and Server

9.12 Gopher

Gopher is a protocol used for information retrieval. It is not as popular as HTTP, but there are still some Gopher servers to which you may wish to provide access. Usually the client machine does not use a specific client for Gopher, because the most popular Web browsers already have support for this protocol.

The server uses tcp/70, and the client can use any nonprivileged port.

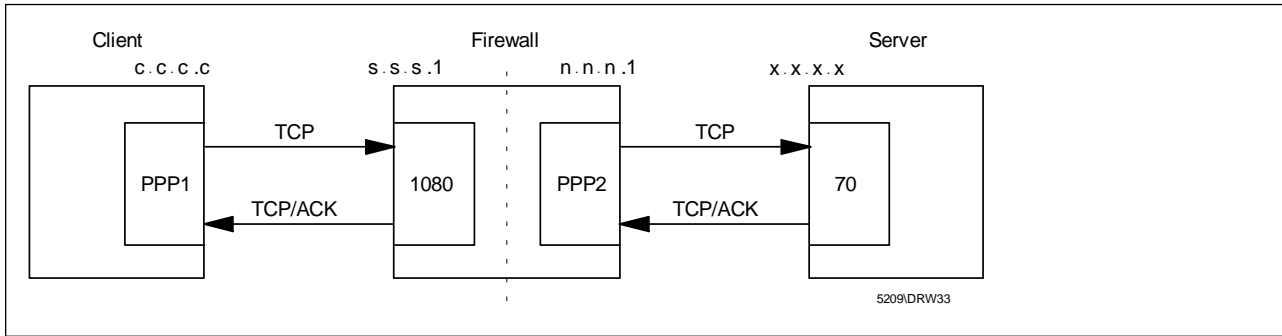


Figure 179. Gopher Using a SOCKSified Client

Two connections are needed for this configuration. The first is the standard SOCKS client connection (see Figure 153 on page 174). The second is as follows (the rules in this connection are not part of the standard set, so they must be manually created).

Connection: Gopher from Firewall to Gopher Server

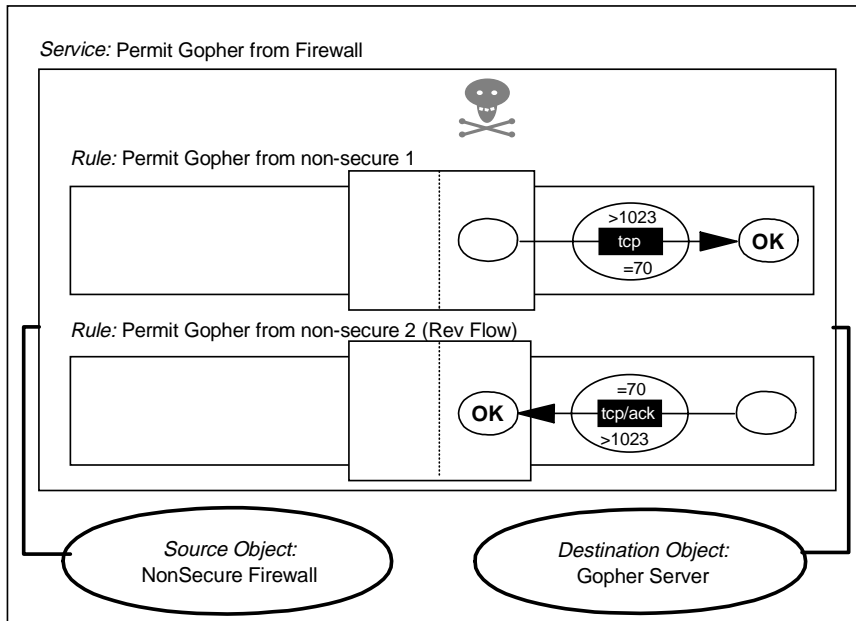


Figure 180. Gopher from SOCKS to Non-secure Server

The equivalent filter definitions are as follows:

```
# Connection from the client to the SOCKS Server
permit s.s.s.s sm.sm.sm.sm s.s.s.1 0xffffffff tcp gt 1023 eq 1080 secure local inbound
permit s.s.s.1 0xffffffff s.s.s.s sm.sm.sm.sm tcp/ack eq 1080 gt 1023 secure local outbound

# Connection from Firewall to the Server in the Non-secure Network
permit n.n.n.1 0xffffffff 0 0 tcp gt 1023 eq 70 nonsecure local outbound
permit 0 0 n.n.n.1 0xffffffff tcp/ack eq 70 gt 1023 nonsecure local inbound
```


9.13 Lotus Notes and Domino

A full description of Lotus Notes firewall configurations can be found in *The Domino Defense: Security in Lotus Notes and the Internet*, SG24-4848. We will describe the major aspects regarding the firewall rule base setup.

A Notes server is normally inside a private enterprise network. Defining rules to access the server is not an issue in that case. If we are using Domino to provide access to Notes databases from the Internet, we recommend that you position the Domino server in the DMZ, as shown in Figure 181. It is put in the DMZ, because a Domino server is, from the point of view of a client on the Internet, just another HTTP server, so we apply the same practices for positioning and protecting it as an HTTP server (9.9, "HTTP - World Wide Web Sessions" on page 192).

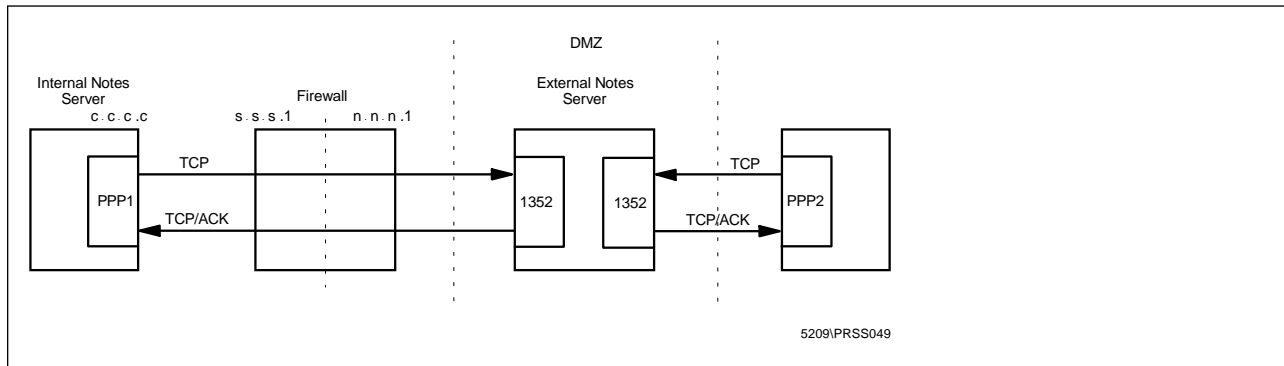


Figure 181. Domino in a DMZ Configuration

To keep the Domino server up to date we use Notes' replication capabilities. Because we only want to allow outgoing Notes connections we have to use the Push-Pull kind of replication, initiated from the secure network. This means that although database updates are replicated in both directions, the replication session is only initiated by the secure Notes server.

Replication and client access to a Notes server are both handled by the same standard TCP protocol on port 1352.

We have two options:

1. Use SOCKS: the internal Notes server must be SOCKSified.

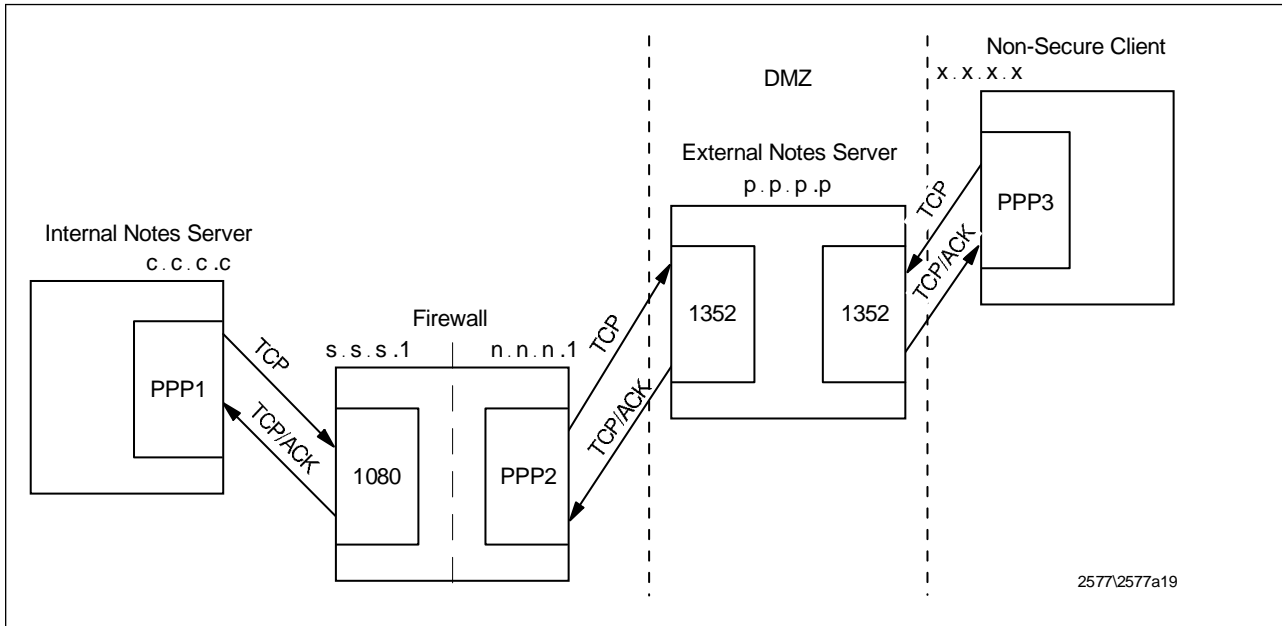


Figure 182. Domino in a DMZ Configuration Using SOCKS

To split up the connection on the firewall we can use a similar setup as with HTTP via SOCKS. For this setup we need to implement the SOCKS client connection, (Figure 172 on page 193) plus the following connection to allow the SOCKS server to communicate with the external Notes server.

Connection: Notes from Firewall to Domino Server

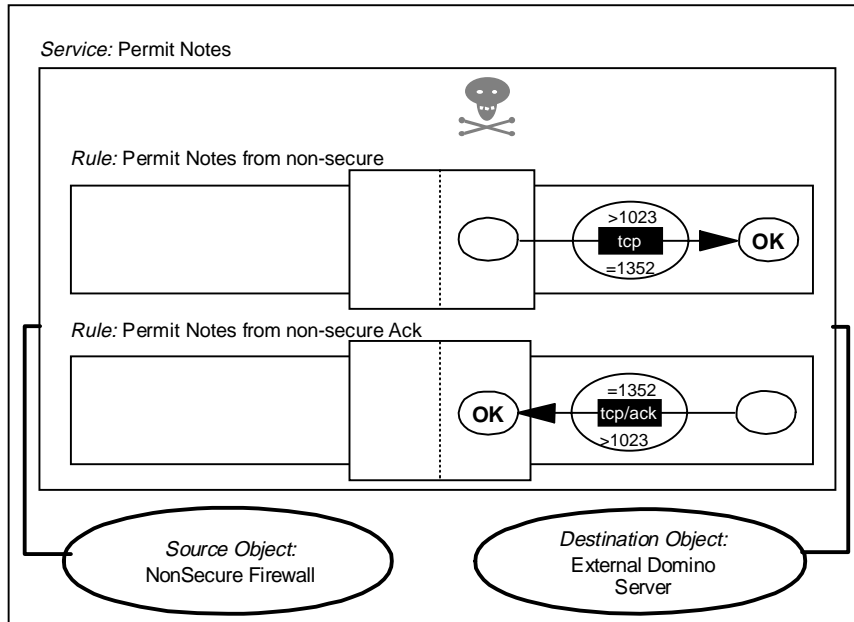


Figure 183. Lotus Notes Using SOCKS to Non-secure Server

The equivalent filter rules are as follows:

```
# Connection from the Internal Notes Server to the SOCKS Server
permit c.c.c.c 0xffffffff s.s.s.1 0xffffffff tcp gt 1023 eq 1080 secure local inbound
permit s.s.s.1 0xffffffff c.c.c.c 0xffffffff tcp/ack eq 1080 gt 1023 secure local outbound

# Connection from the Firewall to the External Notes Server on the DMZ
```

```

permit n.n.n.1 0xffffffff p.p.p.p 0xffffffff tcp gt 1023 eq 1352 nonsecure local outbound
permit p.p.p.p.0xffffffff n.n.n.1 0xffffffff tcp/ack eq 1352 gt 1023 nonsecure local inbound

```

To use this setup you also have to configure your internal Notes server by editing the SOCKS proxy field in the Notes Server document to reflect the address of the firewall. If your version of Notes does not support SOCKS, you can try the tcp_relay setup. In this case, you need to configure the internal Notes server to replicate to the firewall instead of the external Notes server, and configure tcp_relay only in one direction.

2. Allow routed traffic through the firewall

We have to route the replication connection through the firewall. This configuration presents some risks. A cracker in the Internet can run another Notes server which is similar with the internal Notes server and do a replication. Of course, this crackers would still have to overcome the problem of identifying to the internal Notes server. You could change the port number of the external Notes server to present another hurdle to the cracker. You can change either the replication port number or domino access port number.

Still another alternative would be to place the external Notes server in another subnetwork using a firewall third adapter. See Chapter 17, "Configuring the IBM Firewall for Three Adapters" on page 449 for a detail discussion of the firewall third adapter.

9.14 ident

The ident protocol is used to authenticate the user of a connection. The ident client sends a request containing a user ID and information about a TCP port that the user ID appears to be using. The ident server responds positively if the user really is using the port. In practice there are very few platforms that implement the ident server as standard and it has no value on systems that do not require users to log in (such as Windows 3.1, Windows 95 and OS/2).

The server (the machine who's connection is being authenticated) uses tcp/113 and the client (the machine that is trying to authenticate the connection) uses a nonprivileged port.

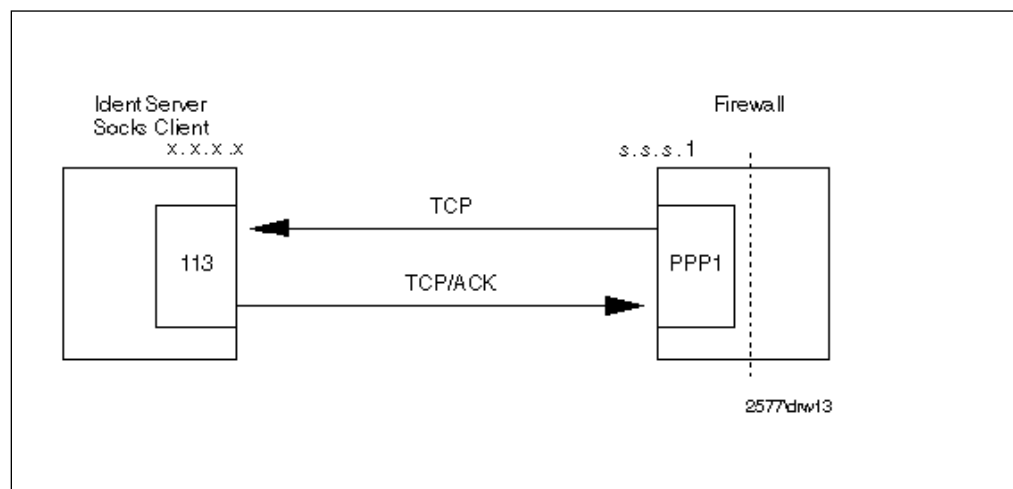


Figure 184. ident, for Authentication from the Firewall to the Secure Network

If you are planning to use SOCKS with user authentication, you should enable this service from the firewall to the secure network. Figure 185 on page 202 shows you this connection.

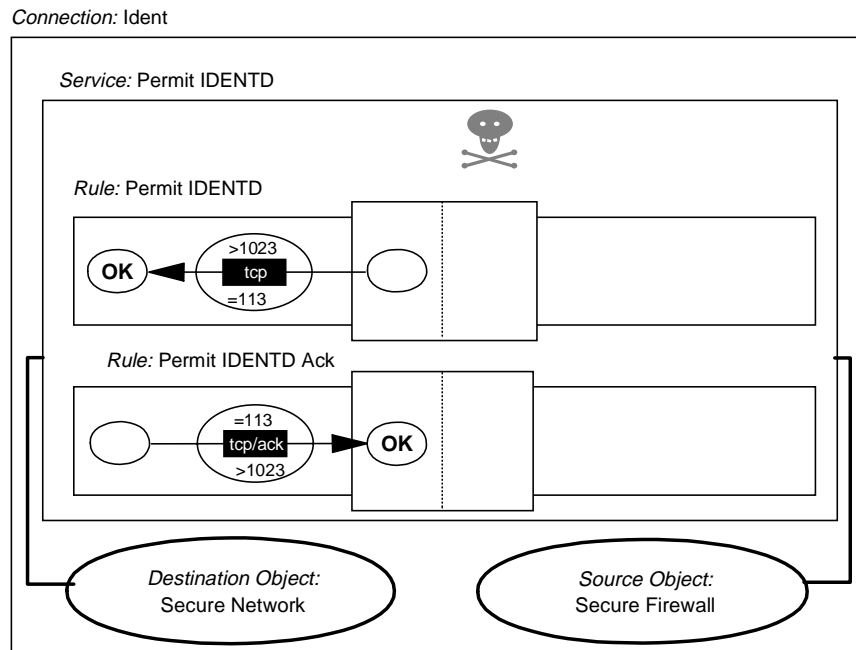


Figure 185. Ident from SOCKS to Secure Network

9.15 Traceroute

Traceroute is a service that is useful in allowing network administrators to track the path that an IP packet is following in order to reach its final destination. It works by sending UDP packets from one high port (port number >1023) to another high port. It selects a free UDP port and starts to send packages to different high ports.

In order to discover the path, it plays some tricky games with the TTL value of the packet (this field must be decremented by routers every time they forward the packet). First it sends a UDP packet with TTL=1, so the first router gets the packet, decrements the TTL field, and then discards the packet because the TTL reached 0. After discarding the packet, the router sends an ICMP TTL exceeded message to the sender, so the sender learns the address of the first hop.

Then it uses a TTL value of two, and it gets the second router address. It keeps getting router addresses with TTL exceeded messages until the packet reaches the destination host. Once the destination host receives the packet, it realizes that it doesn't have any service on the high port, and so it sends an ICMP port unreachable message to the sender.

From this description, you can see that using traceroute involves several UDP packets flowing from the sender to the destination, ICMP TTL exceeded messages flowing from the routers to the sender, and finally an ICMP port unreachable message from the destination to the original sender.

Windows NT Version 4.0 implements the traceroute command using echo request and echo reply messages. The command is tracert.

9.15.1 Traceroute from the Firewall

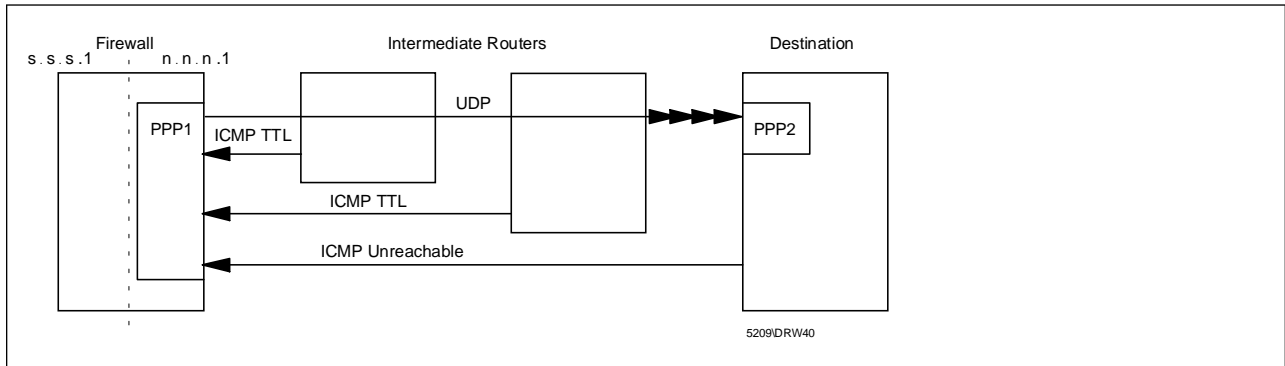


Figure 186. Traceroute Session from the Firewall

This configuration can be safely permitted. In order to do this you must send high UDP packets and accept ICMP TTL and port unreachable messages. Figure 187 shows you the connections needed from the secure network to the firewall secure interface.

Connection: Internal Traceroute

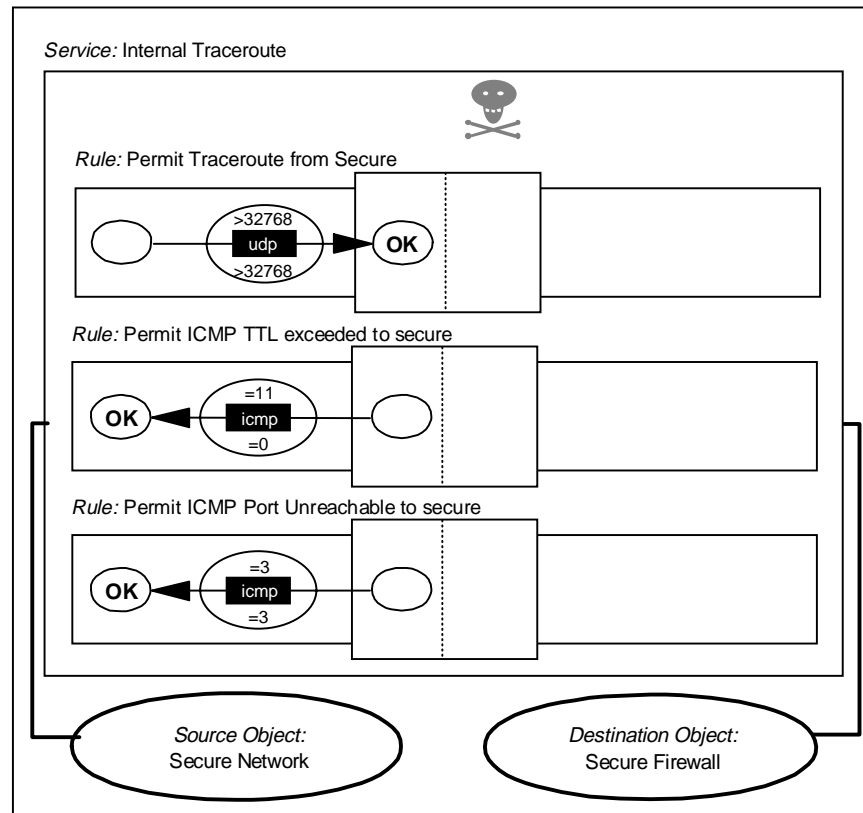


Figure 187. Traceroute from Secure Network to Firewall

Figure 188 on page 204 shows you the connections needed from the firewall non-secure interface to the world.

Connection: External Traceroute

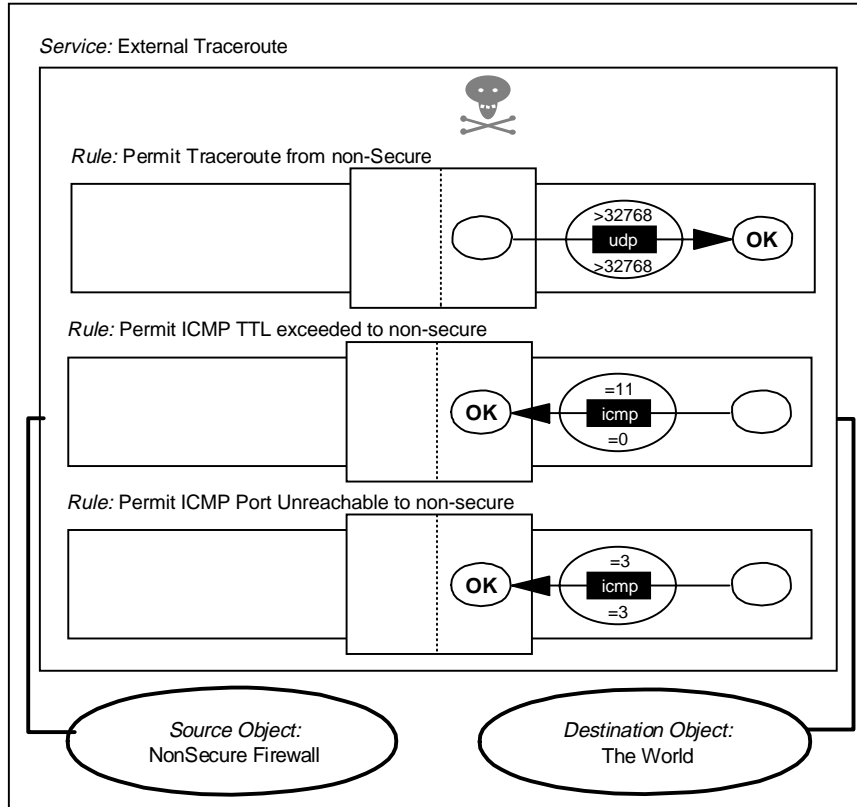


Figure 188. Traceroute from Firewall to Non-secure Network

Note: The ICMP rules are not necessary when the consolidated ICMP connection (see 9.20, “Filtering Specific ICMP Messages” on page 216) is implemented.

The connections implement the following filter rules:

```
# Traceroute to NSN. Outgoing UDP Packets
permit n.n.n.1 0xffffffff 0 0 udp gt 32768 gt 32768 nonsecure local outbound

# Traceroute to SN. Outgoing UDP Packets
permit s.s.s.1 0xffffffff 0 0 udp gt 32768 gt 32768 secure local outbound

# Traceroute. Reply from the Routers
permit 0 0 n.n.n.1 0xffffffff icmp eq 11 eq 0 nonsecure local inbound
permit 0 0 s.s.s.1 0xffffffff icmp eq 11 eq 0 secure local inbound

# Traceroute. Reply from the final node.
permit 0 0 n.n.n.1 0xffffffff icmp eq 3 eq 3 nonsecure local inbound
permit 0 0 s.s.s.1 0xffffffff icmp eq 3 eq 3 secure local inbound
```

9.15.2 Traceroute from Internet to the Firewall

In order to enable this service, you must allow outgoing ICMP port unreachable messages. You may not want to do this because it would be useful to an attacker as a fast way to discover which services you are providing. You can safely allow traceroute from the secure network.

9.16 Network Management Sessions

You may want a network management application (such as TME 10 NetView) to be able to monitor across the secure/non-secure network boundary. Most such applications use the Simple Network Management Protocol (SNMP) for network polling, plus ICMP echo (PING) requests to ascertain whether devices are available or not.

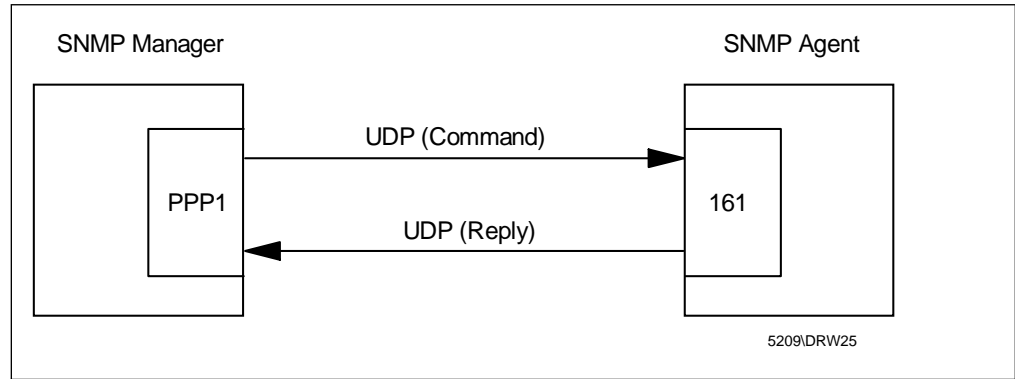


Figure 189. SNMP Manager Querying SNMP Agent

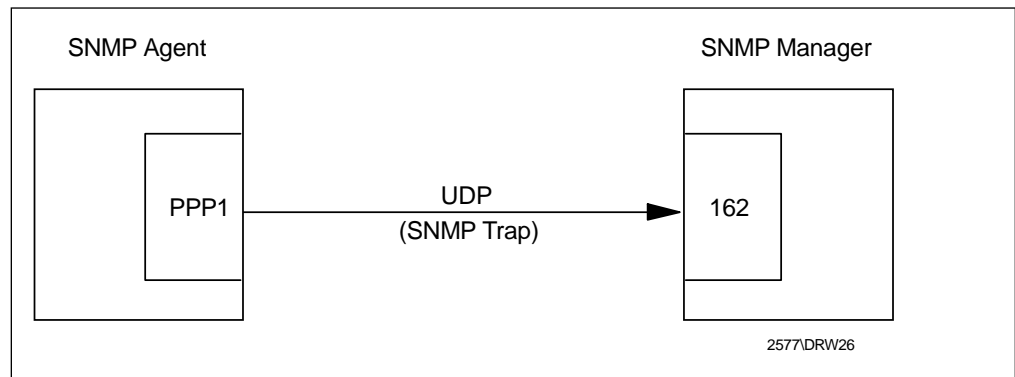


Figure 190. SNMP Agent Notifying SNMP Manager

The following connections will enable an SNMP-based network manager (NV6000) in the secure network to monitor the firewall. The rules used in these connections are not part of the standard set.

Connection: Internal SNMP

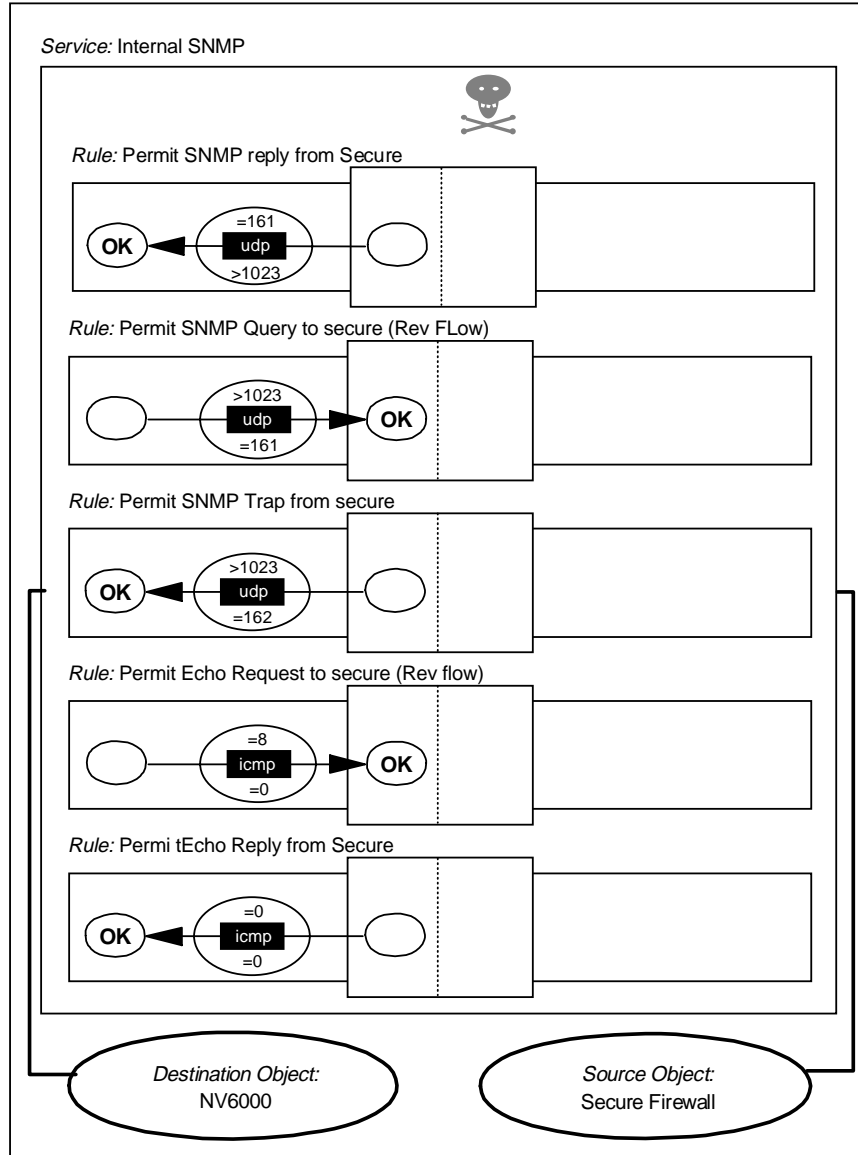


Figure 191. Network Management Sessions

Note: The ICMP rules are not necessary when the consolidated ICMP connection (see 9.20, "Filtering Specific ICMP Messages" on page 216) is implemented.

The first two rules allow SNMP GET, GETNEXT and SET requests to be sent by the manager node to the SNMP agent on the firewall and responses from the firewall. The third rule allows the firewall to send an SNMP trap to the manager node (the manager receives traps by listening on UDP port 162). The final rules allow an ICMP echo request from the network manager to the firewall and an echo reply from the firewall to the network manager.

Note that care should be taken with SNMP if you have any devices that allow update via SNMP SET. Most workstations and routers allow SNMP GET requests only, so there is little damage a cracker can do. However, some devices, notably LAN hubs, allow remote control and configuration functions via SNMP SET requests. SNMP security is imposed by the agent which limits manager access

based on the manager IP address and a community name field. Both of these fields are carried in clear in the UDP packet, so it is not difficult for an attacker to fool the agent into giving him full control.

The following filter rules are implemented by the above connection:

```
# SNMP Get, GetNext and Set from Network Manager to Firewall
permit N.V.6.K 0xffffffff s.s.s.1 0xffffffff udp gt 1023 eq 161 secure local inbound
permit s.s.s.1 0xffffffff N.V.6.K 0xffffffff udp eq 161 gt 1023 secure local outbound

# SNMP Traps from Firewall to Network Manager
permit s.s.s.1 0xffffffff N.V.6.K 0xffffffff udp gt 1023 eq 162 secure local outbound

# Ping from N.V.6.K to Firewall
permit N.V.6.K 0xffffffff s.s.s.1 0xffffffff icmp eq 8 any 0 secure local inbound
permit s.s.s.1 0xffffffff N.V.6.K 0xffffffff icmp eq 0 any 0 secure local outbound
```

9.17 Archie

Archie is a service useful for searching programs in anonymous FTP servers. The Archie servers maintain a database of program names, locations and descriptions.

Archie is a UDP protocol that uses port 1525 for the server and nonprivileged ports for the client. There are several ways in which users can use this protocol, circumventing UDP packets. They can use a WWW gateway, a Telnet client, or access Archie through mail.

In order to use the WWW gateway, users just need HTTP access (see 9.9, “HTTP - World Wide Web Sessions” on page 192). In this case, they will just have to open a Gateway Form Document, like <http://archie.rutgers.edu/archie.html>. (see Figure 192 on page 208).

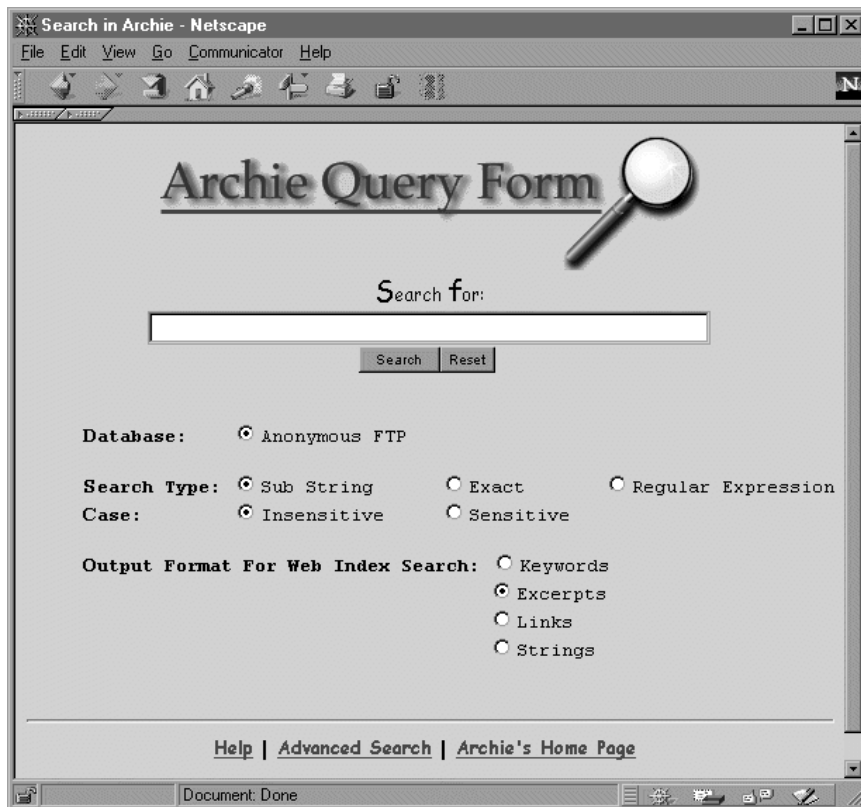


Figure 192. Archie Using an HTTP Gateway

9.18 WAIS

WAIS (Wide Area Information Servers) is a service used to search through large text databases. It uses TCP as a transport layer; servers use port 210, and the clients use any nonprivileged port.

One easy way to provide WAIS service is through an HTTP gateway. The client just selects a URL that provides a WAIS service (for example, <http://www.ai.mit.edu/the-net/wais.html>) and submits the query. In this way, if you are already providing HTTP access you can provide, WAIS access for free (you just need to educate your users).

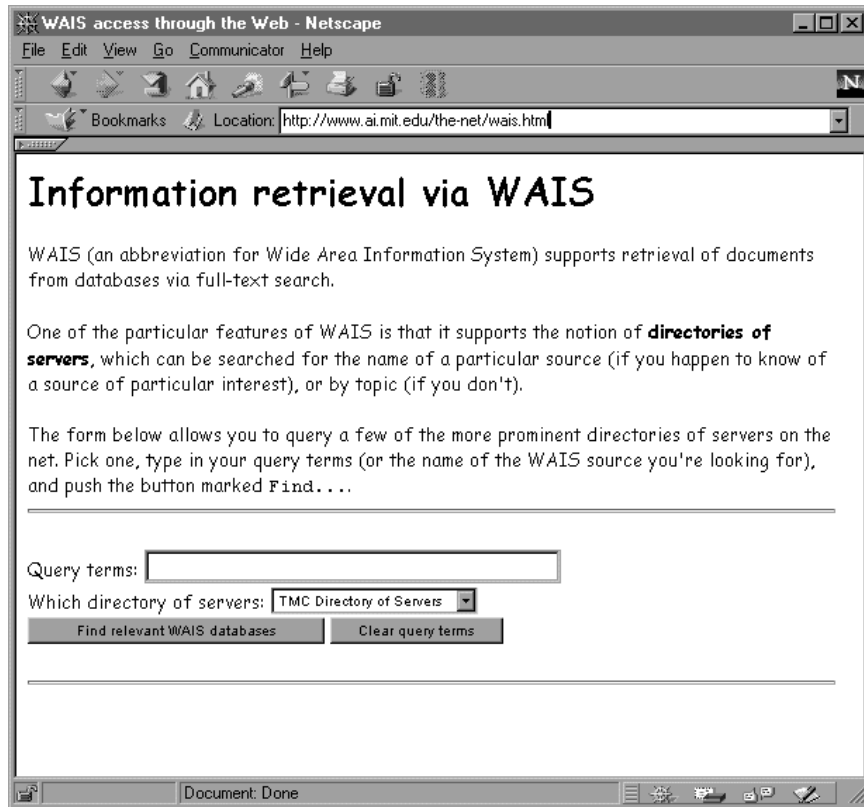


Figure 193. WAIS Using an HTTP Gateway

You could also allow SOCKSified dedicated clients. To permit this you will need the following connection that allows the firewall to contact the WAIS server on port 210 (note that the rules used in this service are not part of the standard set and so must be manually defined).

Connection: WAIS from Firewall to Anywhere

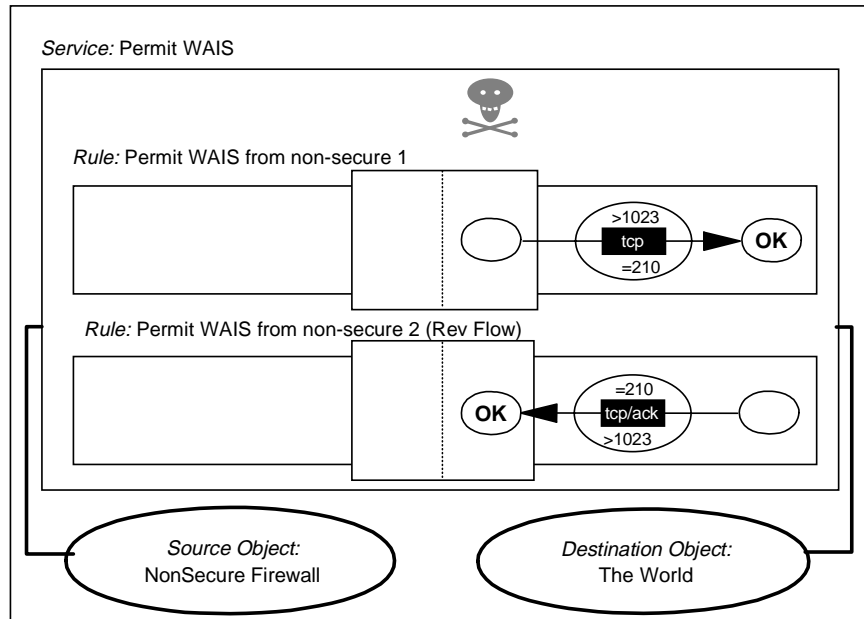


Figure 194. WAIS from SOCKS to the Non-secure Network

```

# Connection from the client to the SOCKS Server
permit s.s.s.s sm.sm.sm.sm s.s.s.s 0xffffffff tcp      gt 1023 eq 1080 secure local inbound
permit s.s.s.s 0xffffffff s.s.s.s sm.sm.sm.sm tcp/ack eq 1080 gt 1023 secure local outbound

# Connection from Firewall to the Server in the Non-secure Network
permit n.n.n.1 0xffffffff 0 0 tcp      gt 1023 eq 210 nonsecure local outbound
permit 0 0 n.n.n.1 0xffffffff tcp/ack eq 210  gt 1023 nonsecure local inbound

```

9.19 HOD: Host On-Demand

Host On-Demand (HOD) is an Internet-to-Host connectivity solution that provides host-based application discovery and access through the WWW. Web users can use HOD from their Java-enabled desktops or Web browsers to access central computer data. For Web-oriented users with new or growing needs for central computer access, HOD provides an alternative to installing a terminal emulation program on desktop.

The basic scenario of HOD that we used in our tests is the client using a Java-enabled browser to access the Web Server as a gateway. From the Web Server, the client can choose to which server the client wants to create a connection. Then, depending on the configuration of HOD, the user will get a Host connection through a separated session. The method we use to get a connection with the Host is *Redirection*. Redirector is a new function in HOD V2.0. It allows you to proxy the telnet session between client and host. So, no client can access the Host directly. And the other advantage is that HOD can act as a gateway to a pool of Hosts.

In this section, we want to set up rules which will enable clients in an external network to access an internal HOD server. The most secure way to do this is through a Virtual Private Network. We also still can use other options which use the strong authentication provided by the IBM eNetwork Firewall for Windows NT

This section only discusses firewall connections, services and rules that are related to the scenarios we tested. To enable those scenarios, we have to create TCP sessions on port TCP/80, TCP/8999 and TCP/12001. HOD uses Web port 80 for Web server, port 8999 for redirector and port 12001 (configurable) for session. The possible scenarios that are covered in this section are:

1. Access an internal HOD server using IP Forwarding
2. Access an internal HOD server using SOCKS

HOD supports several other configurations, like SSL and installation of HOD in the firewall, but we are not covering these possibilities in this document.

9.19.1 HOD Server Using IP Forwarding

Obviously, this scenario is not recommended. There is a security exposure by allowing routed traffic through the firewall. Although only for specific ports, this scenario will enable TCP connection through the firewall. In order to reduce the vulnerability, we assume that the client is located in a trusted network; only specific hosts can access the HOD server.

The configuration is shown in Figure 195 on page 211.

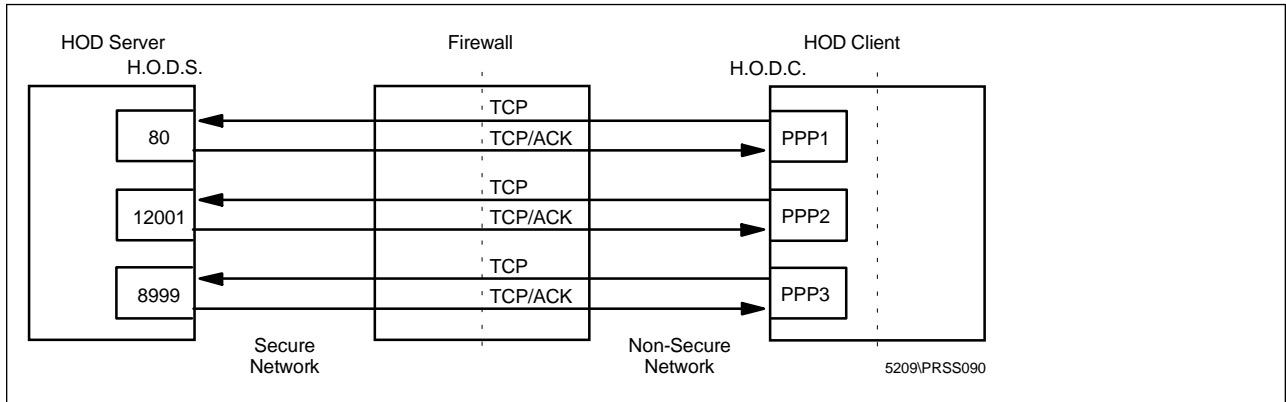
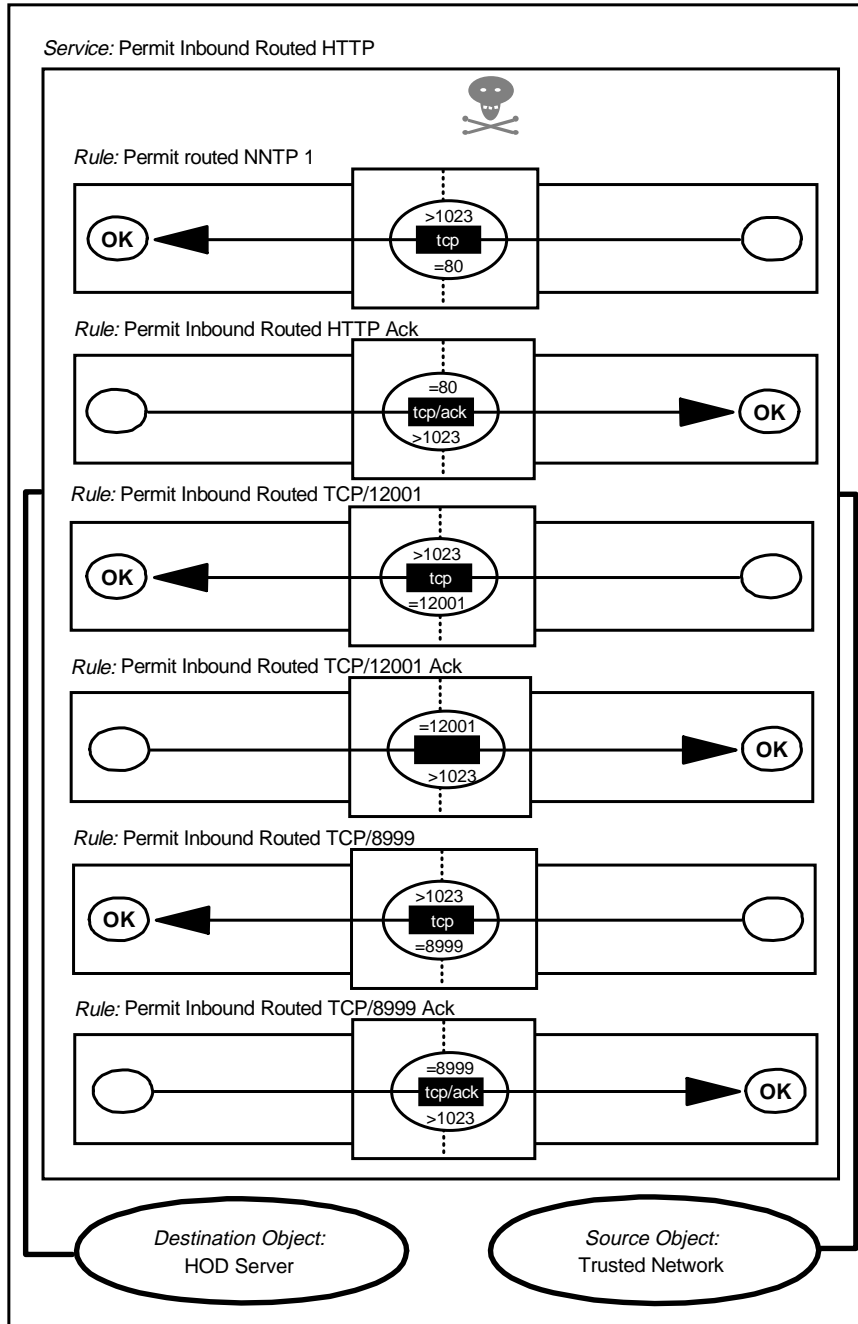


Figure 195. Configuration of HOD Server Using IP Forwarding

The first two rules are used to enable web connection and the other four rules are used for the HOD connection.

Connection: Permit Inbound Routed HOD



5209\PRSS091

Figure 196. HOD Server Using IP Forwarding

The following filter rules are implemented by this connection:

```
#Connection for Inbound Routed HOD Server
permit H.O.D.0 0xffffffff H.O.D.S 0xffffffff tcp gt 1023 eq 80 both route both
permit H.O.D.S 0xffffffff H.O.D.0 0xffffffff tcp/ack eq 80 gt 1023 both route both
permit H.O.D.0 0xffffffff H.O.D.S 0xffffffff tcp gt 1023 eq 12001 both route both
permit H.O.D.S 0xffffffff H.O.D.0 0xffffffff tcp/ack eq 1023 gt 12001 both route both
permit H.O.D.0 0xffffffff H.O.D.S 0xffffffff tcp gt 1023 eq 8999 both route both
permit H.O.D.S 0xffffffff H.O.D.0 0xffffffff tcp/ack eq 1023 gt 8999 both route both
```

9.19.2 HOD Server Using SOCKS

In this scenario, we will separate the connection between the non-secure network and the secure network to make it safer. The following steps are used to set up an Inbound HOD access using SOCKS.

1. Disable **Deny SOCKS to non-secure interface.**

You can do it through Security Policy; see 9.23, “Using the Security Policy Panel” on page 219.

2. Create rules, services and connections.

The rules, services and connections for inbound HOD access using SOCKS are shown in Figure 198 on page 214 and Figure 199 on page 215.

3. Create new SOCKS rules.

You have to add a new SOCKS rule in order to allow TCP connections to the HOD server on port 12001 and 8999. See 13.7.2, “Configuring SOCKS Services” on page 326 for information about creating a SOCKS rules. You can also choose one of the authentication methods that SOCKS V5 offers.

4. Install SOCKS V5 client software in the remote user.

In order to use the strong authentication provided by SOCKS V5, we have to use a SOCKS V5-compliant client software. At the time of writing this document, there are no browsers that support SOCKS V5. You have to install a SOCKS V5 product that SOCKSifies the whole TCP/IP stack of the client. We choose AutoSOCKS from Aventail Inc. for this purpose; see 21.2, “Aventail AutoSOCKS” on page 521 for more information about AutoSOCKS.

The following configuration is used for HOD using SOCKS.

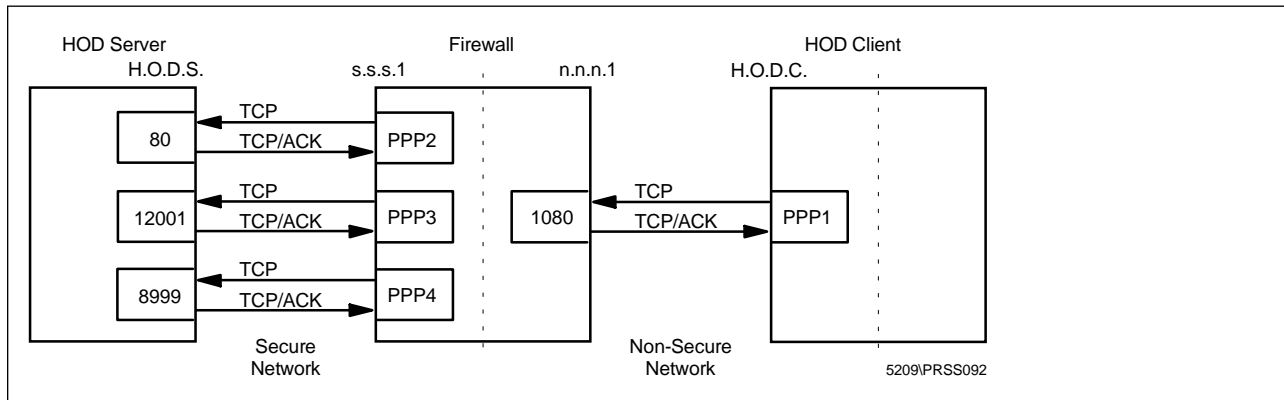
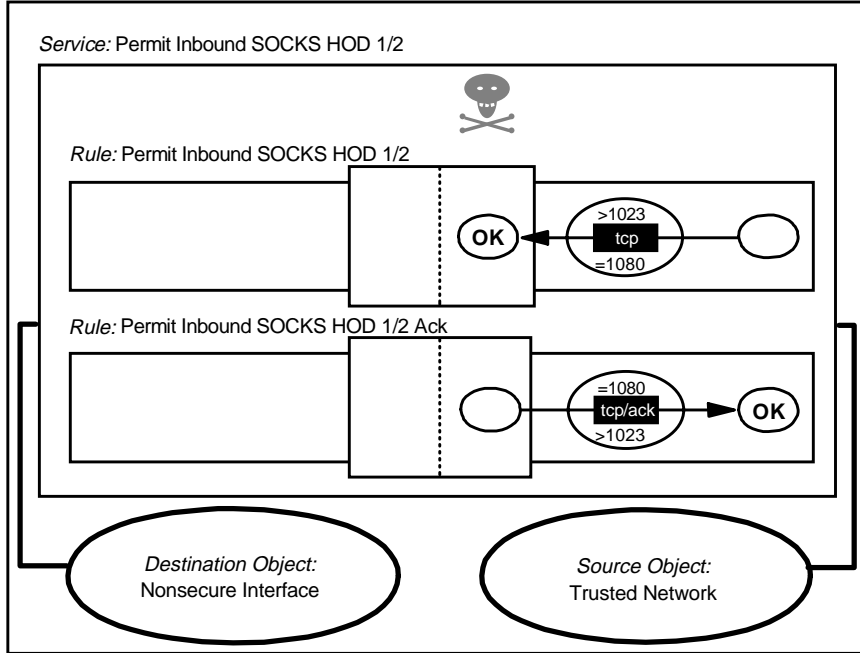


Figure 197. Configuration of HOD Server Using SOCKS

There is one connection for accessing the SOCKS server on port TCP/1080 from the non-secure network, and three other connections to enable the traffic between the firewall and the HOD server.

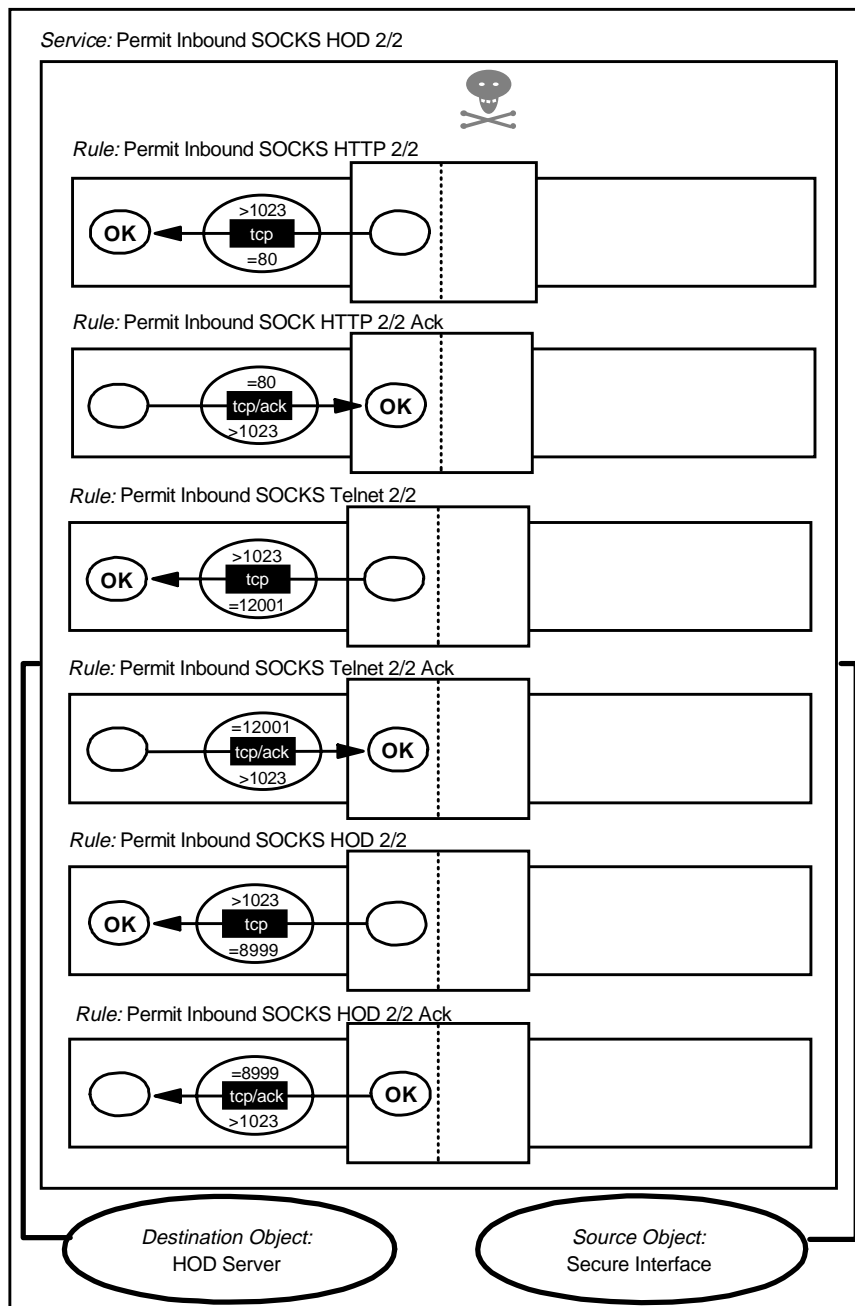
Connection: Permit Inbound SOCKS HOD 1/2



5209\PRSS093

Figure 198. Connection from Trusted Network to SOCKS Server

Connection: Permit Inbound SOCKS HOD 2/2



5209\PRSS094

Figure 199. Connection from SOCKS Server to HOD Server

These connections generate the following filter rules:

```
#Connection from Trusted Network to Non-secure Interface
permit H.O.D.0 0xffffffff n.n.n.1 0xffffffff tcp    gt 1023 eq 1080 nonsecure local inbound
permit n.n.n.1 0xffffffff H.O.D.0 0xffffffff tcp/ack eq 1080 gt 1023 nonsecure local outbound

#Connection from Secure Interface to Web Server
permit s.s.s.1 0xffffffff H.O.D.S 0xffffffff tcp    gt 1023 eq 80 secure local outbound
permit H.O.D.S 0xffffffff s.s.s.1 0xffffffff tcp/ack eq 80 gt 1023 secure local inbound

#Connection from Secure Interface to TCP/8999
permit s.s.s.1 0xffffffff H.O.D.S 0xffffffff tcp    gt 1023 eq 8999 both route both
permit H.O.D.S 0xffffffff s.s.s.1 0xffffffff tcp/ack eq 23 gt 8999 both route both
```

```
#Connection from Secure Interface to Redirector Port
permit s.s.s.1 0xffffffff H.O.D.S 0xffffffff tcp gt 1023 eq 12001 both route both
permit H.O.D.S 0xffffffff s.s.s.1 0xffffffff tcp/ack eq 23 gt 12001 both route both
```

9.20 Filtering Specific ICMP Messages

At the start of this chapter we suggested that a simple but robust approach to ICMP is just to block all ICMP messages. Many of the ICMP messages may be misused by an attacker, but there are also many which are benign, and which contribute to the smooth running of your applications.

In 7.2, “An Introduction to ICMP Packets” on page 129, we described the different types of ICMP packets at some length and included a recommended filtering strategy for each one. The following connection will implement those recommendations.

9.20.1 Consolidated ICMP Connection

Connection: Consolidated ICMP

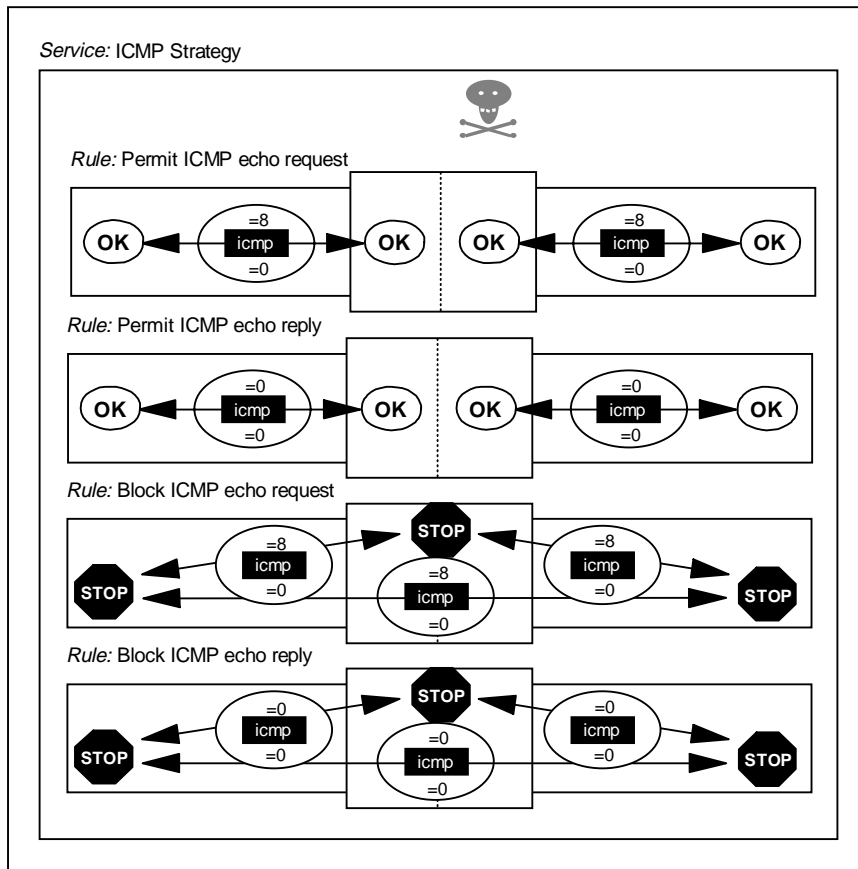
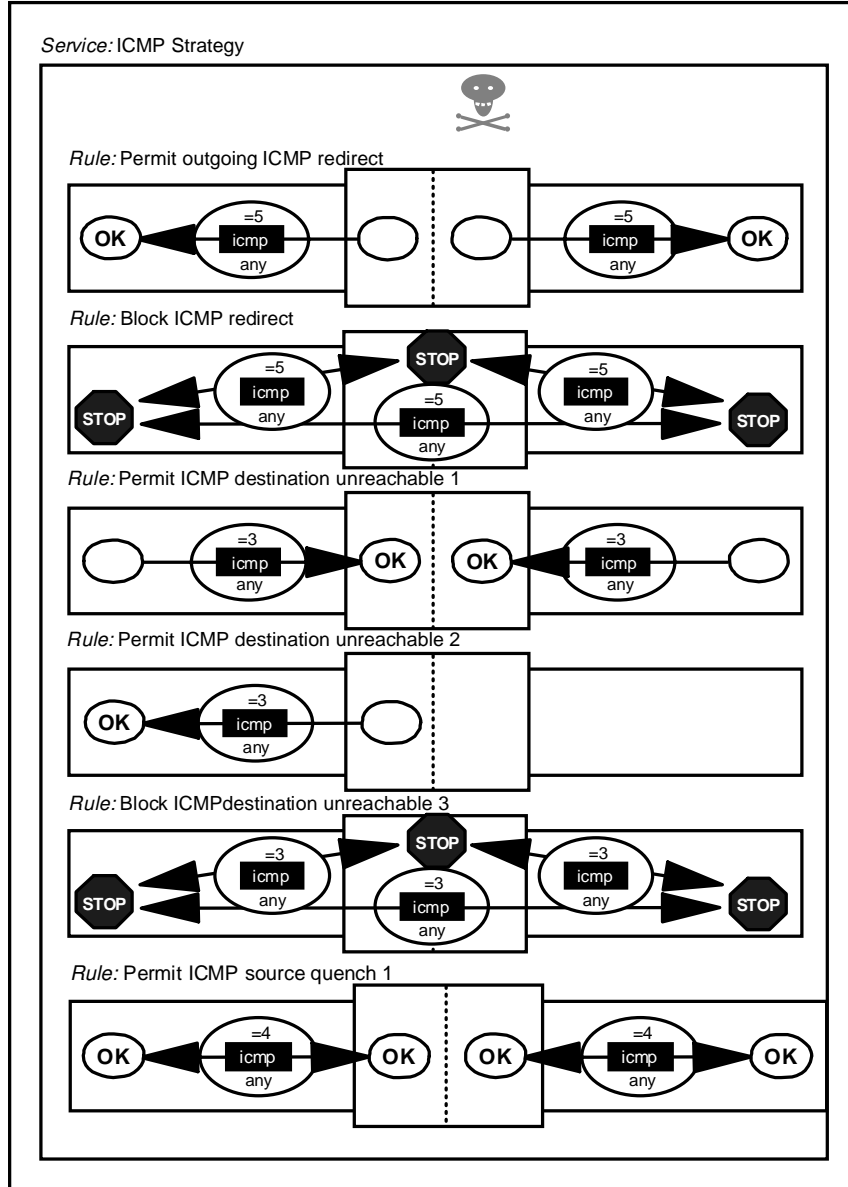
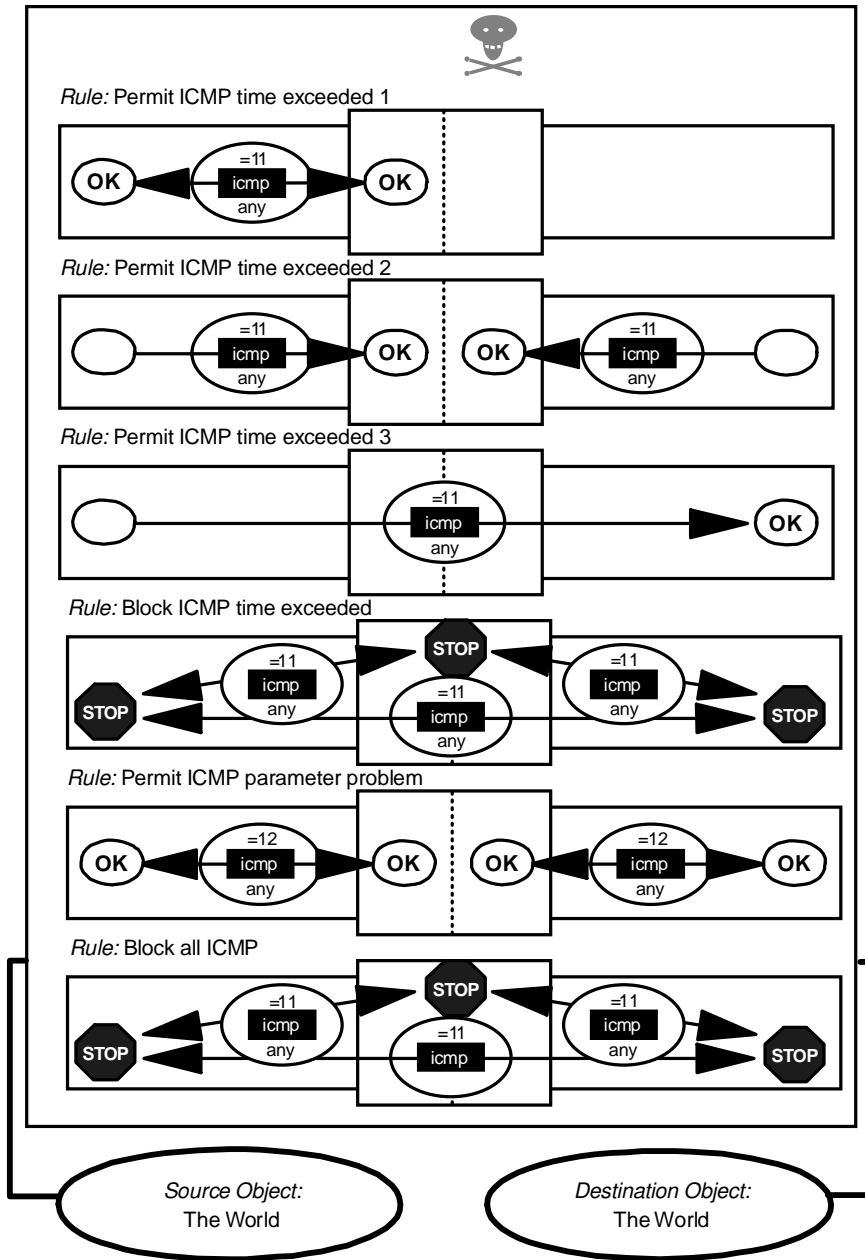


Figure 200. Consolidated ICMP Control Connection (Part 1)



5209\SNGPPIX40

Figure 201. Consolidated ICMP Control Connection (Part 2)



5209\SNGPPIX41

Figure 202. Consolidated ICMP Control Connection (Part 3)

9.21 Other Protocols

As the Internet is constantly evolving, there are always new protocols, so we want to stress the last but most important rule of the firewall. This default rule is always added by IBM eNetwork Firewall for Windows NT:

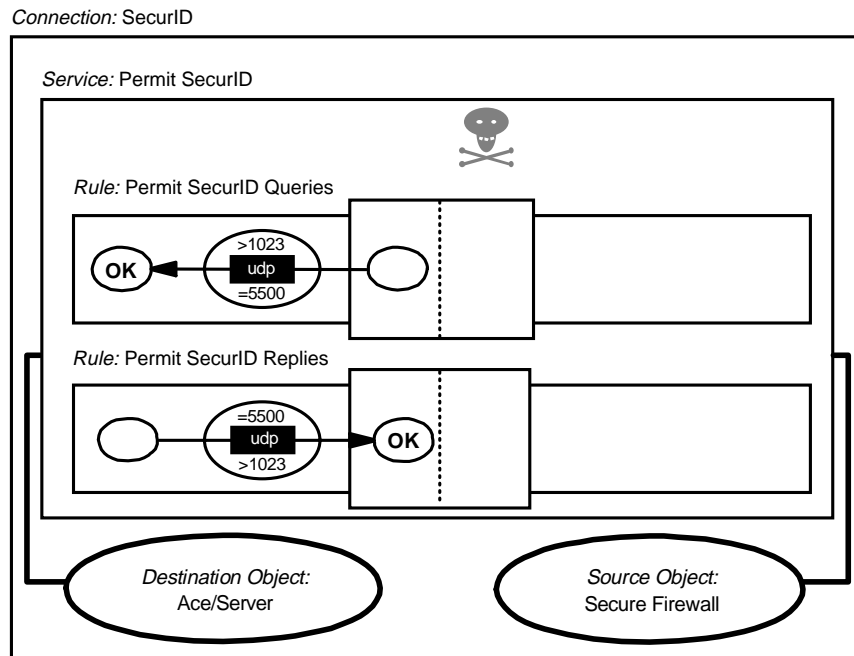
```
# Everything that is not explicitly allowed is denied.
deny 0 0 0 0 all any 0 any 0 both both both
```

9.22 SecurID by Security Dynamics

Security Dynamics provides a strong authentication method by providing tokens that generate one-time passwords. See 21.5, “SecurID Authentication” on page 563 for details.

The SecurID agent is located in the firewall; the ACE/Server that authenticates the one-time passwords will probably be located in the secure network. The communication between the agent and the manager uses UDP, with the manager listening to port 5500.

We need to allow this traffic to flow from the firewall to the secure network. For this purpose, the following connection must be created:



5209\PRSS095

Figure 203. SecurID

The IBM Firewall has predefined services to allow this kind of traffic. See 21.5.2.6, “SecurID Connection” on page 574 for details.

9.23 Using the Security Policy Panel

Some of the rules mentioned in this chapter can be set up using the Security Policy panel. If you select Security Policy from the main GUI panel, you get the screen shown in Figure 204 on page 220.

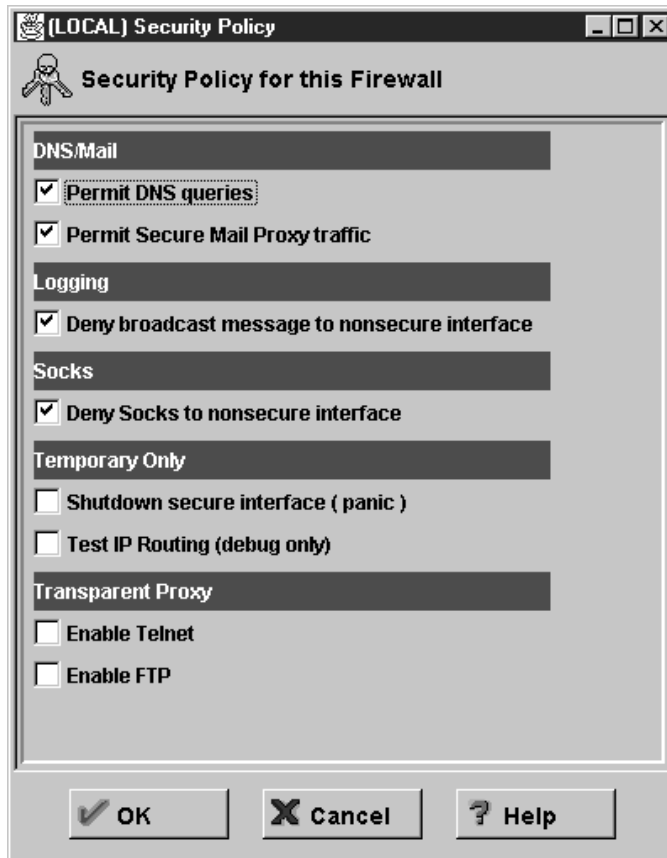


Figure 204. Security Policy Panel

From this panel you can set up the rules for DNS, broadcast and SOCKS similar to the ones mentioned in this chapter.

The advantage of using the Security Policy panel over the method described before, is that it is standard and provides a common way of setting up basic filters in all the IBM eNetwork Firewall for Windows NT installations. The disadvantages are that it is not possible to control the rule placement and the rules could be too general or too specific.

The Security Policy panel can also be used to:

- Close the non-secure interface in case you suspect the firewall is under attack.
- Open the firewall: no restriction to the traffic. Use only for testing as this opens the door of the secure network to everybody.
- Enable/disable the Telnet and/or FTP transparent proxies.

Chapter 10. Remote Administration Configuration

The Configuration Client is a separate component from the IBM Firewall code itself. While it is installed together with the base firewall code on the firewall machine, it can also be loaded on other Windows 95 or Windows NT machines on the network. From these remote clients, it is possible for administrators to configure various aspects of the firewall.

Remote configuration is useful in large networked environments where an administrator may not be in the local vicinity of the firewall. An example would be a multinational corporation with multiple firewalls spread out across different parts of the globe. If a local administrator is unavailable at the time a firewall needs reconfiguration, an administrator from another location could log on to the firewall and take the required steps.

There are two types of administrators on the firewall: the primary firewall administrator, and ordinary firewall administrators. Primary administrators have complete control over the firewall settings, while ordinary administrators can only control areas assigned to them. Ordinary administrators can only log in to remote clients. This leads to another reason for remote configuration: delegation of administrative tasks.

Remote configuration is not without its risks, however, as the communication between the client and the firewall could possibly be picked up using a network sniffer. To circumvent this, SSL encryption is included as an option for the link between the client and the server.

To activate remote configuration, you must set up the Configuration Server on the firewall and the Configuration Client on another machine in the network. Additionally, you might want to create firewall administrator accounts as well.

10.1 Setting Up the Configuration Server

The Configuration Server is the Configuration Client's interface to the firewall. The Configuration Server processes requests from the Configuration Client; it runs on the firewall machine and can handle requests from Configuration Clients that are on either the local or remote machine. The Configuration Server runs as the Windows NT service called IBM Firewall Configuration Server on the firewall machine. It is started automatically.

10.1.1 The Configuration Service Port Number

The Configuration Server's port number is specified in the `c:\winnt\system32\drivers\etc\services` file. The particular entry is `ibmfwrccs`, located somewhere near the bottom of the file. The port number defaults to 1014, but you can change this value for added security. See 10.5, "Changing Port Numbers" on page 240 for more information.

Note: Throughout this chapter we assume that the files are in drive C:.

10.1.2 The Client Configuration File fwcfgsrv

The Configuration Server is initially set up to accept only requests from Configuration Clients on the local machine. Initial requests are not encrypted. The configuration file which specifies these settings is c:\Program Files\IBM\Firewall\config\rscfile.cfg.

A listing follows below.

```
# The local= parameter indicates whether or not remote configuration
# can occur on any machine or just from the local machine.
# local=yes - means configuration can occur only on the local machine;
#             this is the default
# local=no  - means configuration can occur from any machine
local=yes

# The encr= parameter indicates what type of encryption should
# occur for data being sent to and from the remote configuration
# server.

# encr=none - means that no encryption will occur; this is the default
# encr=ssl  - means that ssl encryption will occur
encr=none

# The sslfile= parameter indicates the full path name of
# ssl keyfile which contains the key certificate to be used
# for SSL encryption. The keyfile's associated stash file must
# be placed in the same directory as the keyfile.
#
# Note: Use the mkkf utility to create the keyfile and its
# associated stash file.
sslfile=C:\Program Files\IBM\Firewall\config\fwkey.kyr
```

Figure 205. Listing of rscfile.cfg

This file may be viewed or modified using the command `fwcfgsrv`, which is executed from the command prompt. The syntax for `fwcfgsrv` is given below.

fwcfgsrv gives the options available with the command.

```
C:\>fwcfgsrv

Usage:

fwcfgsrv cmd=list

fwcfgsrv cmd=change
          [localonly=yes|no]
          [encryption=none|ssl]
          [sslfile=ssl_file_name]

See the IBM Firewall Reference Guide for more information.
```

Figure 206. Running fwcfgsrv

fwcfgsrv cmd=list displays the information contained in the configuration file.

```
C:\>fwcfgsrv cmd=list
localonly = yes
encryption = none
sslfile =
```

Figure 207. Running *fwcfgsrv cmd=list*

fwcfgsrv cmd=change modifies the configuration variables `localonly`, `encryption`, and `sslfile`. Examples will be given in the following sections.

10.1.3 Enabling Remote Configuration Client Connections

To enable the firewall for remote configuration, first modify the `rcsfile.cfg` to tell the firewall to accept remote Configuration Clients. Then, activate the filter rules for remote configuration so that the firewall does not reject the connections.

10.1.3.1 Changes in the Configuration File

To enable the firewall for remote configuration, simply modify `rcsfile.cfg` to reflect the changes. You can do this using `fwcfgsrv`.

1. Open a Command Prompt window.
2. From the command prompt, execute the command `fwcfgsrv cmd=change localonly=no`.

```
C:\>fwcfgsrv cmd=change localonly=no
Command completed successfully.
```

Figure 208. Running *fwcfgsrv cmd=change*

3. Verify the change by running `fwcfgsrv cmd=list`.

```
C:\>fwcfgsrv cmd=list
localonly = no
encryption = none
sslfile =
```

Figure 209. Verifying the Changes with *fwcfgsrv cmd=list*

10.1.3.2 Activating Filter Rules for Remote Configuration

Next, enable the firewall to accept connections from remote Configuration Clients at port 1014. Remote configuration connections may come from either the secure network or the non-secure network. The example that follows is the setup for remote configuration from the secure network.

1. Log on to the local firewall machine.
2. Create a network object for the administration client. Open the **Network Objects** document and double-click on **<NEW>** of type Single.

3. Create a host object as shown in Figure 210 on page 224.

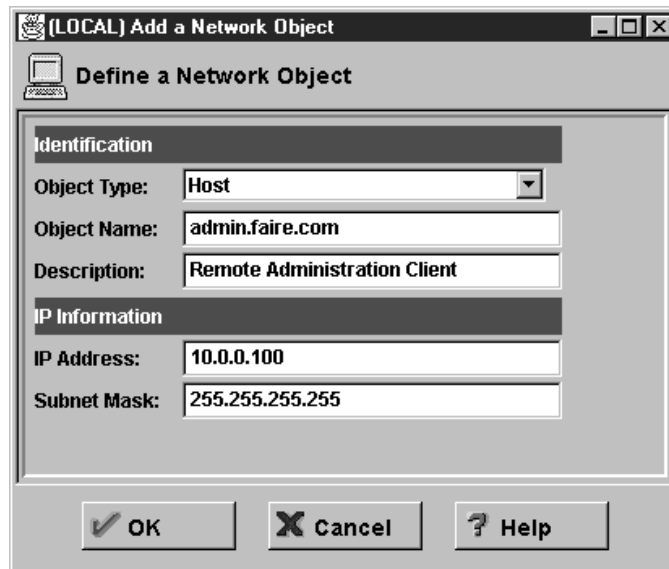


Figure 210. Creating a Network Object Definition for an Administration Client

4. On the Navigation Tree, open the **Traffic Control** folder and double-click on the **Connection Setup** document. This will bring up the Connections List dialog box.
5. Double-click on **<NEW>** to add a new connection. This will bring up the Add a Connection dialog box.
6. Create a new object with the parameters shown in Figure 211 on page 225.

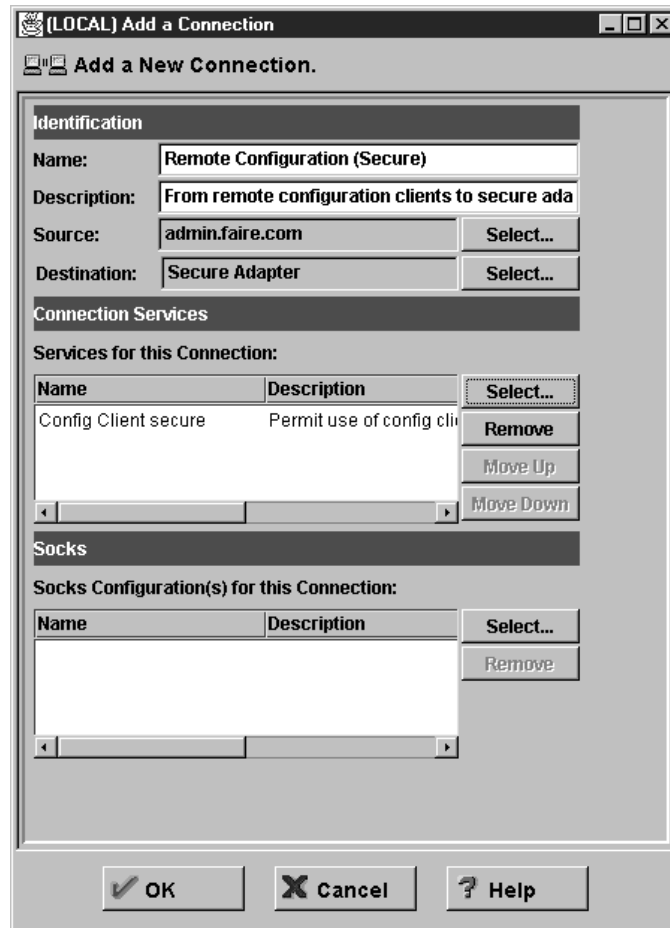


Figure 211. Adding a New Connection

The source for the connection will be the specified remote administration client, admin.faire.com.

The destination will be the Secure Adapter network object.

In the Connection Services list, add the predefined service Config Client secure.

7. After you close the Add a Connection dialog, you will go back to the Connections List dialog. Click on **Activate** to bring up the Connection Activation dialog.
8. Select the **Regenerate Connection Rules and Activate** radio button and click on **Execute**.

If you want to enable the firewall to accept connections from remote Configuration Clients from the non-secure network, follow the steps given above, except change *Source*, *Destination*, and the *Connection Service* accordingly. There is a predefined service called Config Client non-secure that is appropriate for this.

If you are planning to configure the firewall from the non-secure network, we strongly suggest that you enable SSL encryption. See 10.2, “Enabling Encrypted Remote Configuration” on page 226 for more information.

10.2 Enabling Encrypted Remote Configuration

Both the firewall and the firewall Configuration Client support remote configuration over an SSL-encrypted channel. This secure SSL network connection requires that you have a key file that the firewall uses to encrypt the communications with the client. This SSL key file must also be specified in the rcsfile.cfg configuration file.

10.2.1 Creating a Key File with mkkf

The firewall comes with its own SSL key generation utility called mkkf. The steps to create the key file are detailed in Chapter 6 of *IBM eNetwork Firewall for Windows NT Reference Version 3 Release 3*. A sample session is in 10.2.2, "A Sample mkkf Session."

10.2.2 A Sample mkkf Session

You must be logged on to an account with administrative rights on the firewall to run the mkkf utility .

In the example below, user response is highlighted in boldface.

```
C:\Program Files\IBM\Firewall\config>mkkf

.....
MKKF Key Manager
Copyright IBM Corp. 1996
All Rights Reserved

    Currently Selected Key Ring: (none)

N - Create New Key Ring File
O - Open Key Ring File
X - Exit

Enter a command: n
Enter a name for the key ring file, or press Enter for keyfile.kyr.
fwkey.kyr

Key Ring Menu
    Currently Selected Key Ring: fwkey.kyr

N - Create New Key Ring File
O - Open Key Ring File
S - Save Key Ring File
A - Save Key Ring as Another File
P - Set Password for Key Ring File
C - Create Stash File for Key Ring File
R - Receive a Certificate into a Key Ring File
W - Work with Keys and Certificates
X - Exit

Enter a command: w

Key Menu
    Currently Selected Key Ring: fwkey.kyr
    Selected Key Entry: (none)
```

L - List/Select a Key To Work With
C - Create a New Key and Certificate Request
I - Import a Key From an Armored Key File
X - Exit This Menu

Enter a command: **c**
Enter password to use for the key file: *********
Enter the password again for verification: *********
Should the password expire?
Enter Y for yes or N for No:
n
Password successfully set.
Press ENTER to continue

Choose Certificate Request Type Menu
S - PEM Certificate Request Format
P - PKCS10 Certificate Request Format
C - Cancel

Enter a command: **s**

Compose PEM Certificate Request Menu

Current Certificate Information

Key Name: (none)
Key size: 0
Server Name: (none)
Organization: (none)
Organizational Unit: (none)
City/Locality: (none)
State/Province: (none)
Postal Code: (none)
Country: (none)

M - Modify the Certificate Request Fields
R - Ready To Create Key and Certificate Request
C - Cancel

Enter a command: **m**
Enter a name to use for the key entry:

Firewall Key

1: 508
2: 512
3: 768
4: 896
5: 1024

Enter the number corresponding to the key size you want: **2**

Enter the server's fully qualified TCP/IP domain name
or press ENTER by itself to leave the field blank.
blue.faire.com

Enter Organization Name for the certificate
or press ENTER by itself to leave the field blank.
Faire.Com

Enter Organizational Unit Name for the certificate
or press ENTER by itself to leave the field blank.
Network Connectivity Group

Enter Locality/City Name for the certificate
or press ENTER by itself to leave the field blank.
Makati

Enter State/Province Name for the certificate
or press ENTER by itself to leave the field blank.
State/Province must be at least three characters long
Manila

Enter Postal Code for the certificate
or press ENTER by itself to leave the field blank.
1226

Enter Country Code for the certificate
or press ENTER by itself to leave the field blank.
Country code must be exactly two characters long.
PH

Compose PEM Certificate Request Menu

Current Certificate Information
Key Name: Firewall Key
Key size: 512
Server Name: blue.faire.com
Organization: Faire.Com
Organizational Unit: Network Connectivity Group
City/Locality: Makati
State/Province: Manila
Postal Code: 1226
Country: PH

M - Modify the Certificate Request Fields
R - Ready To Create Key and Certificate Request
C - Cancel

Enter a command: **r**
Enter file to store the certificate request in: **fwkey.cert**
Creating Private Key....
Private key was successfully created.
Creating certificate request.....
Certificate request was successfully created
Adding new key to key file.
The new key and certificate request were created successfully.
Press ENTER to continue

Key Menu
Currently Selected Key Ring: fwkey.kyr
Selected Key Entry: Firewall Key

L - List/Select a Key To Work With
S - Show Information about Selected Key
D - Delete Selected Key
C - Create a New Key and Certificate Request

- I - Import a Key From an Armored Key File
- E - Export Selected Key To an Armored Key File
- F - Make Selected Key the Default Key for this Key Ring
- T - Mark Selected Key as a Trusted Root
- R - Create A Certificate Request For Selected Key
- X - Exit This Menu

Enter a command: **f**

Currently selected key: Firewall Key

Are you sure you want to make this key the default?

Enter Y for yes or N for No:

y

Key was made the default key.

Press ENTER to continue

Key Menu

Currently Selected Key Ring: fwkey.kyr

Selected Key Entry: Firewall Key

- L - List/Select a Key To Work With
- S - Show Information about Selected Key
- D - Delete Selected Key
- C - Create a New Key and Certificate Request
- I - Import a Key From an Armored Key File
- E - Export Selected Key To an Armored Key File
- F - Make Selected Key the Default Key for this Key Ring
- T - Mark Selected Key as a Trusted Root
- R - Create A Certificate Request For Selected Key
- X - Exit This Menu

Enter a command: **x**

Key Ring Menu

Currently Selected Key Ring: fwkey.kyr

- N - Create New Key Ring File
- O - Open Key Ring File
- S - Save Key Ring File
- A - Save Key Ring as Another File
- P - Set Password for Key Ring File
- C - Create Stash File for Key Ring File
- R - Receive a Certificate into a Key Ring File
- W - Work with Keys and Certificates
- X - Exit

Enter a command: **r**

Enter file name or press ENTER for Cert.txt.

fwkey.cert

This is a self-signed certificate. Add it to key file?

Enter Y for yes or N for No:

y

Certificate added to key ring.

Press ENTER to continue

Key Ring Menu

Currently Selected Key Ring: fwkey.kyr

N - Create New Key Ring File
O - Open Key Ring File
S - Save Key Ring File
A - Save Key Ring as Another File
P - Set Password for Key Ring File
C - Create Stash File for Key Ring File
R - Receive a Certificate into a Key Ring File
W - Work with Keys and Certificates
X - Exit

Enter a command: **c**
Stashed password file saved to fwkey.sth
Press ENTER to continue

Enter a command: **x**
Key ring file has been changed. Save?
Enter Y for yes or N for No:
y

Keyring saved to fwkey.kyr
Press ENTER to continue

C:\Program Files\IBM\Firewall\config>

Running mkkf will create three files in the config directory: the SSL fwkey, the key ring, and the stash file. In this example, the files are fwkey.cert, fwkey.kyr, and fwkey.sth, respectively. The firewall uses fwkey.kyr to encrypt the files.

You can change the name of the key file, but it must always have the extension .kyr.

10.2.3 Changes in the Configuration File

To enable the firewall for encrypted remote configuration, modify rcsfile.cfg to reflect the change and point to the SSL key file. You can do this using fwcfgsrv. The steps shown below use the files generated in the previous section as an example.

1. Open a Command Prompt window.
2. From the command prompt, execute the command `fwcfgsrv cmd=change encryption=ssl sslfile="c:\program files\ibm\firewall\config\fwkey.kyr"`. You must fill in the complete path for the sslfile variable.
3. Verify the change by running `fwcfgsrv cmd=list`.
4. Open the Services control panel and stop and restart the IBM Firewall Configuration Server service.

SSL or Bust!

Once you've enabled SSL, all your logins must be SSL-encrypted. This includes local logins on the firewall machine.

10.3 Creating Administrators

There are two types of administrators on the IBM Firewall: the primary firewall administrators, and ordinary firewall administrators (otherwise known simply as firewall administrators.)

Primary firewall administrators are users who are members of the Administrators group of Windows NT. Any administrator-level account can log on to a local or remote Configuration Client and make system-wide changes to the firewall. Needless to say, there should only be one primary firewall administrator account on the firewall machine.

Firewall administrators are accounts created on the firewall program itself, not on the Windows NT user database. The primary firewall administrator creates these firewall administrators and assigns one or more tasks to them. These tasks include creation, modification, or deletion of one or more of the firewalls attributes.

The primary firewall administrator specifies which of these firewall attributes the delegated administrator can change. The administrator has authority only over the assigned areas, and no other. However, firewall administrators cannot create, modify, or delete other administrators; this function is reserved for the primary firewall administrator.

Firewall administrators have all the attributes of a proxy user (see 13.1.4, “Configuring Proxy Users” on page 296).

10.3.1 The Primary Administrator Account

As mentioned in the section above, a primary firewall administrator is any member of the Administrators group of the Windows NT user database of the firewall machine. On the firewall, this account (or accounts, in the case of the ill-advised situation where you have more than one administrator-level user in the Windows NT database) maps to the default administrator account, `fwdfadm`.

To explore the attributes of `fwdfadm`, log on to the Configuration Client and open the **Users** document in the navigation tree. In the user list, you will see the `fwdfadm` account. Double-click on **fwdfadm**. This will open its properties dialog, the same dialog for proxy users and firewall administrators. There are three panels here: General, Password, and Administration. For proxy users, the Administration panel will be inactive.

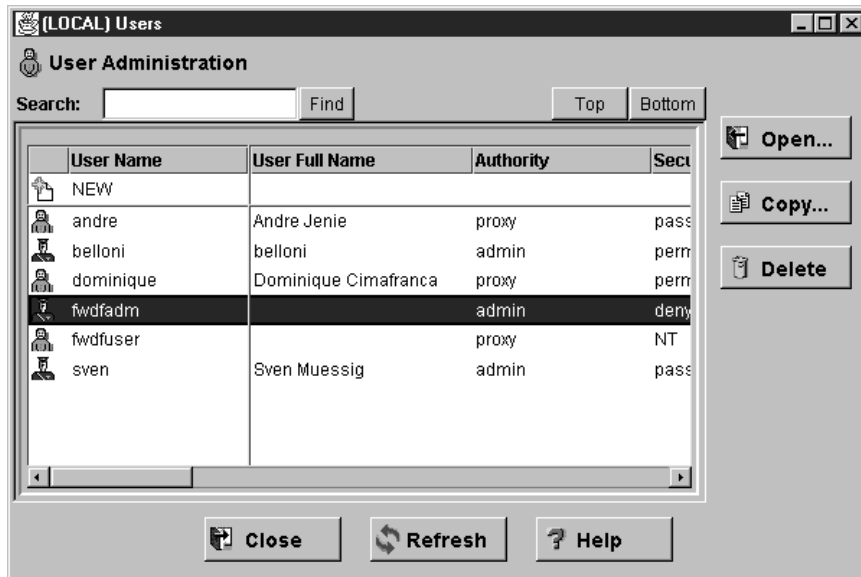


Figure 212. fwdfadm, the Primary Firewall Administrator Account

10.3.1.1 The General Panel

The General panel will define the authority level and authentication methods for fwdfadm. The only two attributes of concern will be the authentication methods for Secure Administration and Non-Secure Administration. Do not change any of the other attributes: firewall administrators should not, in principle, be allowed to use the SOCKS and proxy services.

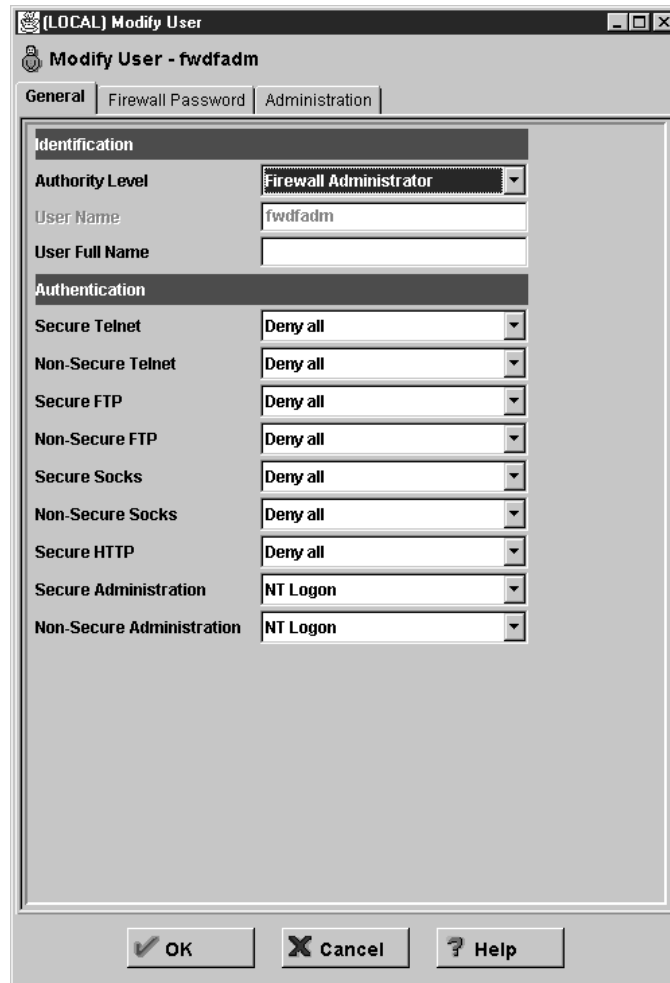


Figure 213. Modifying fwdfadm

The default value for both the Secure Administration and Non-Secure Administration authentication methods is NT Logon. This is what ties the primary firewall administrator's logon password of the Configuration Client to the password in the Windows NT user database.

There will be five other possible options: Deny all, SecurID Card, User-Supplied 1, User-Supplied 2, and User-Supplied 3. 13.1.3, "User Authentication Schemes" on page 295 explains these and other options in greater detail.

Instead of using the NT logon password for authenticating the primary administrator, you might want to use a SecurID card. If you have a SecurID infrastructure set up (see 21.5, "SecurID Authentication" on page 563) you can change the authentication method for the administrator accordingly.

If you do not intend to allow the primary firewall administrator to configure remotely from the non-secure network, the the authentication mode for **Non-Secure Administration** to Deny all.

Set the authentication mode for Secure Administration only to a method which you are absolutely sure you can support. When you log on locally, you are always considered to be doing so from a secure environment, so this is the method you will use. For obvious reasons, never ever set this value to Deny all.

If for some reason this value does get set to Deny all, you will not be able to log on to the firewall Configuration Client. To correct this, go into the C:\Program Files\IBM\Firewall\config directory and edit the file fwusrdb.cfg. Change the line fwsecadmin = deny to fwsecadmin = NT. This enables you to log in to the firewall using the NT password. An even better way to accomplish this change is by using the fwuser command.

10.3.1.2 The Firewall Password Panel

The settings in this panel will have no effect on any aspect of the primary firewall user account. They are used rather to define password characteristics of proxy users and secondary administrators who authenticate via passwords supplied to the firewall directly. See 10.3.2, “Creating a Firewall Administrator Account” on page 235 and 13.1.4, “Configuring Proxy Users” on page 296.

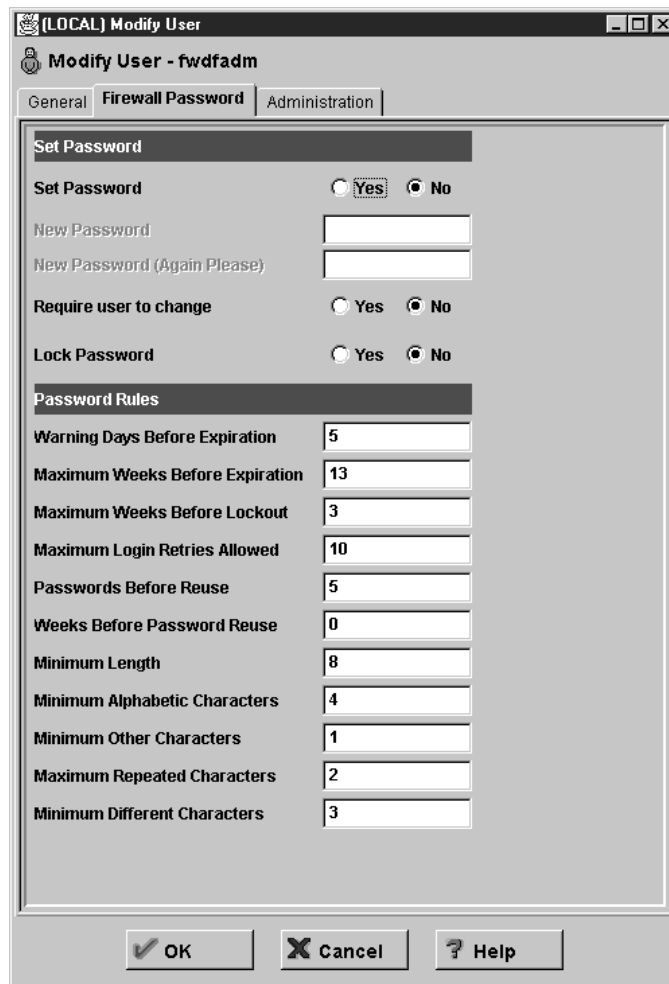


Figure 214. The Firewall Password Panel

10.3.1.3 The Administration Panel

The attributes on this panel are explained in greater detail in 10.3.3, “Administration Functions” on page 236. For the primary firewall administrator, all these attributes should be enabled.

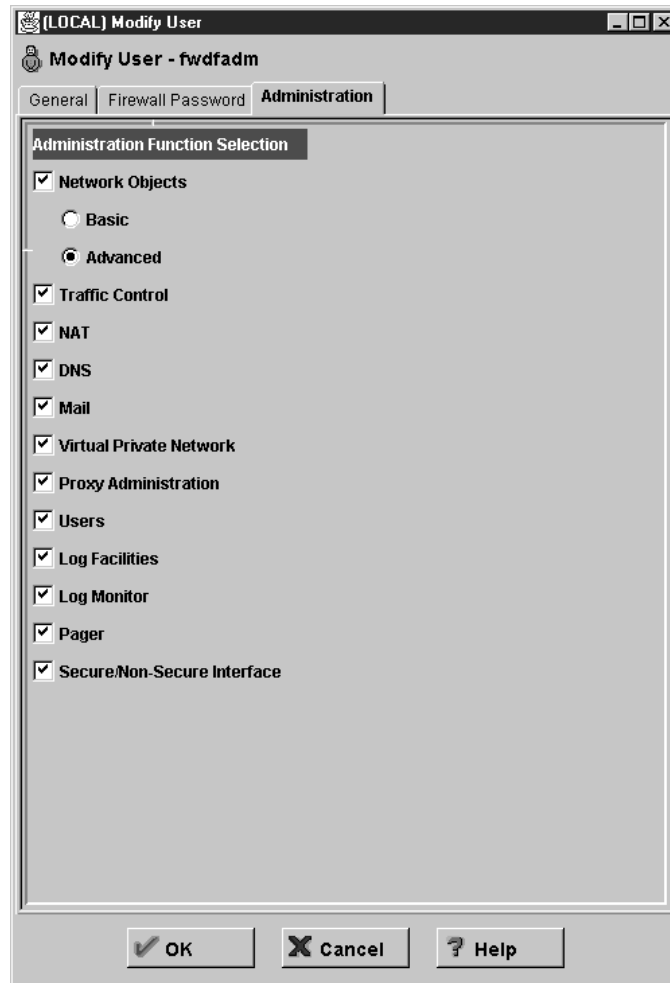


Figure 215. The Administration Panel

10.3.2 Creating a Firewall Administrator Account

Creating a firewall administrator account follows a similar process to creating a proxy user (see 13.1.4, “Configuring Proxy Users” on page 296.)

1. Log on to the Configuration Client and open the **Users** document in the navigation tree.
2. Double-click on **<NEW>**. This will open a properties dialog similar to the dialog for the primary firewall administrator. As before, there are three panels here: General, Password, and Administration. Please see Figure 213 on page 233 for reference.
3. The Authority Level field determines the type of user. Select **Firewall Administrator**. Note that the tab for the Administration panel becomes active.
4. Enter a user name for the administrator, as well as a full user name.
5. Select the authentication method for Secure Administration and Non-Secure Administration. The options available here will be:
 - Deny all** No access.
 - Permit all** No authentication needed.

Firewall Password. The administrator is prompted for a password from the firewall user database (not from the NT user database). This password is specified in the Password panel.

NT Logon The user ID and password will be validated against the local NT user database or against the Windows NT domain database.

SecurID Card Authentication is done using a Security Dynamics SecurID card or pinpad card. Choose this option only if you have an SecurID infrastructure set up.

User-Supplied x Authentication is supplied by the user. Choose this option only if you have set up your own custom authentication methods properly.

As mentioned before, firewall administrator accounts should not be used to access the SOCKS or proxy services. Leave the authentication methods for these services as Deny all.

6. If you selected **Firewall Password**, click on the **Firewall Password** tab to reveal the password attributes panel. Set the password here, as well as other password rules. Please see 13.1.3, "User Authentication Schemes" on page 295 for more information on the available options. Please see Figure 214 on page 234 for reference.
7. Click on the **Administration** tab to set the firewall functions that this administrator will be allowed to configure. (See 10.3.3, "Administration Functions" for an explanation of each).
8. Click on **OK** to add the administrator.

10.3.3 Administration Functions

The functions that a firewall administrator can control are selected from the Administration panel of the User properties dialog. Please see Figure 215 on page 235 for reference.

An explanation of each of the options, and what they control:

Network Objects This option indicates whether this firewall administrator is allowed to configure network objects. If **Network Objects** is selected, decide if you would like this user to have either **Basic** or **Advanced** access to the network objects.

Basic This user cannot modify or delete existing Network Objects. This user can, however, copy or add new network objects.

Advanced All Network Objects functions are available to this user.

Traffic Control This option indicates whether this firewall administrator is allowed to configure filter rule functions.

NAT This option indicates whether this firewall administrator is allowed to configure network address translation.

DNS This option indicates whether this firewall administrator is allowed to configure domain name services.

Mail This option indicates whether this firewall administrator is allowed to configure mail services.

Virtual Private Network This option indicates whether this firewall administrator is allowed to configure virtual private networks.

Proxy Administration This option indicates whether this firewall administrator is allowed to configure proxy services.

Users This option indicates whether this firewall administrator is allowed to define users.

Log Facilities This option indicates whether this firewall administrator is allowed to configure the system logs.

Log Monitor This option indicates whether this firewall administrator is allowed to configure the log monitor function.

Secure/Non-Secure Interface This option indicates whether this firewall administrator is allowed to configure the security status of the interfaces on the firewall.

Pager This option indicates whether this firewall administrator is allowed to set up the pager.

10.4 Setting Up the Configuration Client

Setting up the remote Configuration Client is a trivial task compared to setting up the server. On the firewall installation media, you will have two types of clients available: one for Windows NT and Windows 95, and another for AIX. Both client types can actually be used to configure the IBM eNetwork Firewall for Windows NT remotely. In this section, we will only discuss how to set up the Windows NT/95 Configuration Client.

1. Load the IBM eNetwork Firewall for Windows NT installation media into the remote machine's CD-ROM drive and move to the NT40 directory. Locate the SETUP.EXE file for the client associated with the language you want to install in. Run this file. An InstallShield program will come up.



Figure 216. Installing the Firewall Configuration Client

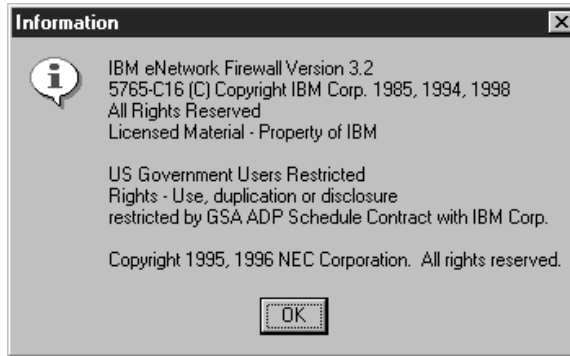


Figure 217. License Information

2. The installation will ask you if you want to install Netscape Navigator 4.04. This is only for reading the help files. If you already have a frames-enabled browser, you do not need to install Navigator.



Figure 218. Prompt for Installing Netscape Navigator

3. The next screen will show you a list of components to install: the firewall client itself, the report utilities, and the SOCKS watcher utility. If this is a first-time installation, then you must install the firewall client. The report utilities and the SOCKS watcher are optional.

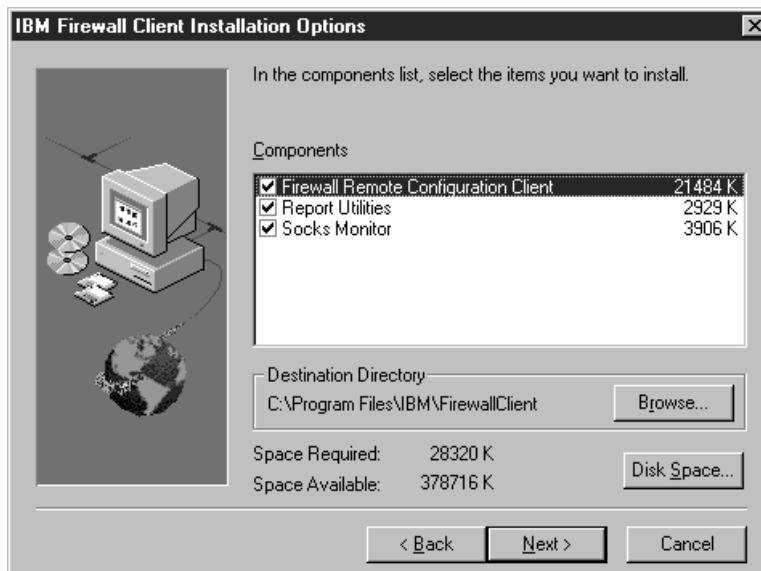


Figure 219. Selecting the Components to Install

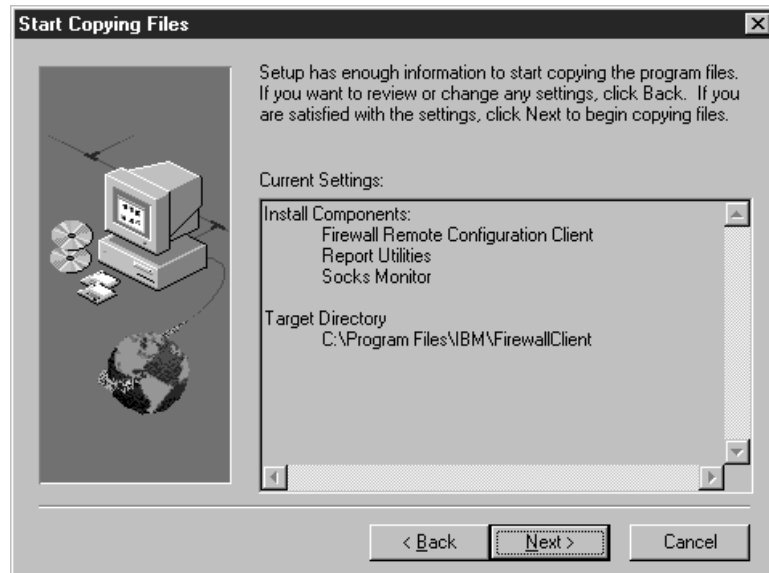


Figure 220. The Components to Install

4. After selecting the components, the installation proceeds with no more user queries. It adds the IBM Firewall program group in your task bar. Reboot the machine after installation.

You start the Configuration Client as you would on the local firewall machine. However, there will be some changes in the options during logon.

- Select **Remote** as the Logon Type. The Host Name field will become active.
- In the Host Name field, type in the host name or IP address of the firewall. If you use the host name, it must be properly registered in the DNS database or in the hosts file of the system on which the Configuration Client is running.
- If you've enabled the server for SSL encrypted communications, select **SSL** from the Encryption drop-down list. Otherwise, leave it at None.
- If you've changed the Configuration Client port number at the server, change the value in the Port Number field. Otherwise, leave it at 1014.
- As usual, select **Host** as the Mode for logon.

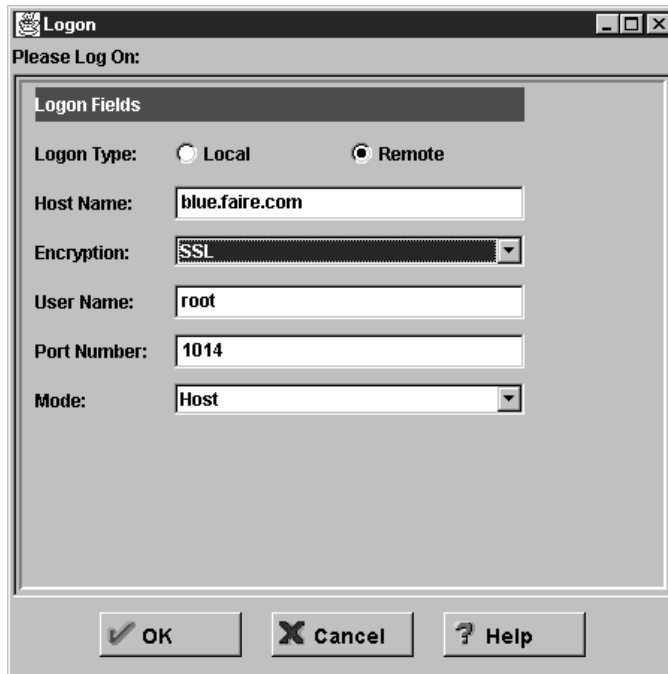


Figure 221. Logging on to the Firewall Remotely

Starting the Configuration Client from the Command Prompt

You can start the Configuration Client from the command line by executing the command file `C:\Program Files\IBM\FirewallClient\cfgcli\gui\ibmfw.bat`. `ibmfw.bat` invokes the Java run-time environment program `jre.exe` and the `FirewallApp` class, which is the primary code for the client. You'll also find some starting options for the client here, but you should have no need to change them.

10.5 Changing Port Numbers

If you are truly paranoid (a good thing to be for firewall administrators), you may want to change the port number used by the remote Configuration Client. To do this:

1. Stop the Configuration Server. Bring up the Services control panel and highlight **IBM Firewall Configuration Server**. Click on **Stop**.
2. Edit the file `c:\winnt\system32\drivers\etc\services` using any text editor. Change the port number for `ibmfwrsc` to some other value. Retain the `tcp` setting. Save the file after you've made the changes.
3. On the Services control panel, highlight the **IBM Firewall Configuration Server** and click on **Start**.

If you make this change, you will also have to create new rules and connections to reflect the new port number. See 10.5.1, "Creating New Rules and Connections" on page 241 for more information.

10.5.1 Creating New Rules and Connections

If you modify the port number for remote configuration, the predefined services Config Client secure and Config Client non-secure will not work! You must create new rules, services, and connections for remote configuration.

1. Open the **Rules** document in the **Traffic Control**→**Connection Template** folder of the Configuration Client navigation tree. This will open the Rules List dialog.
2. In the list, you will find four predefined rules relating to remote configuration: Remote Config Ack out non-secure, Remote Config Ack out secure, Remote Config in non-secure, and Remote Config in secure. All of these rules are set to use port 1014.

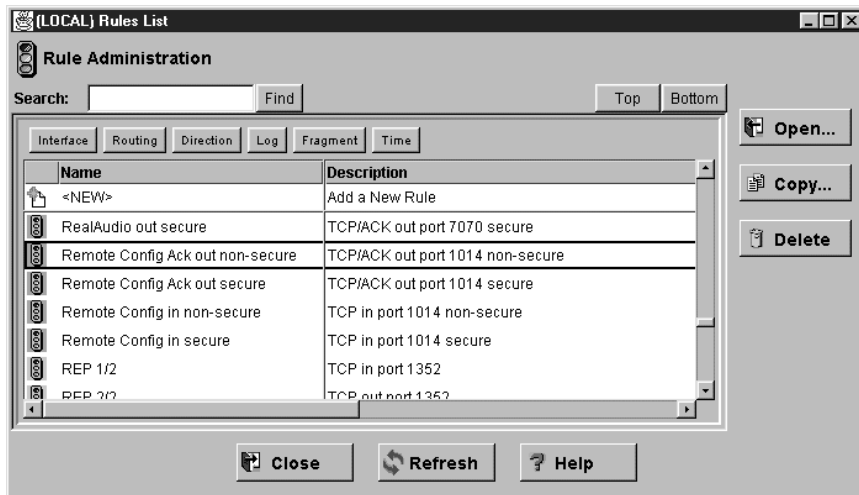


Figure 222. Existing Remote Configuration Rules

3. Create new rules by copying the existing ones. Highlight an existing rule and click on **Copy...** In the Copy IP Rule dialog, change the existing port setting 1014 in either their Source Port or Destination Port so that these fields reflect the new port number you've selected. All other fields and selections are still valid.

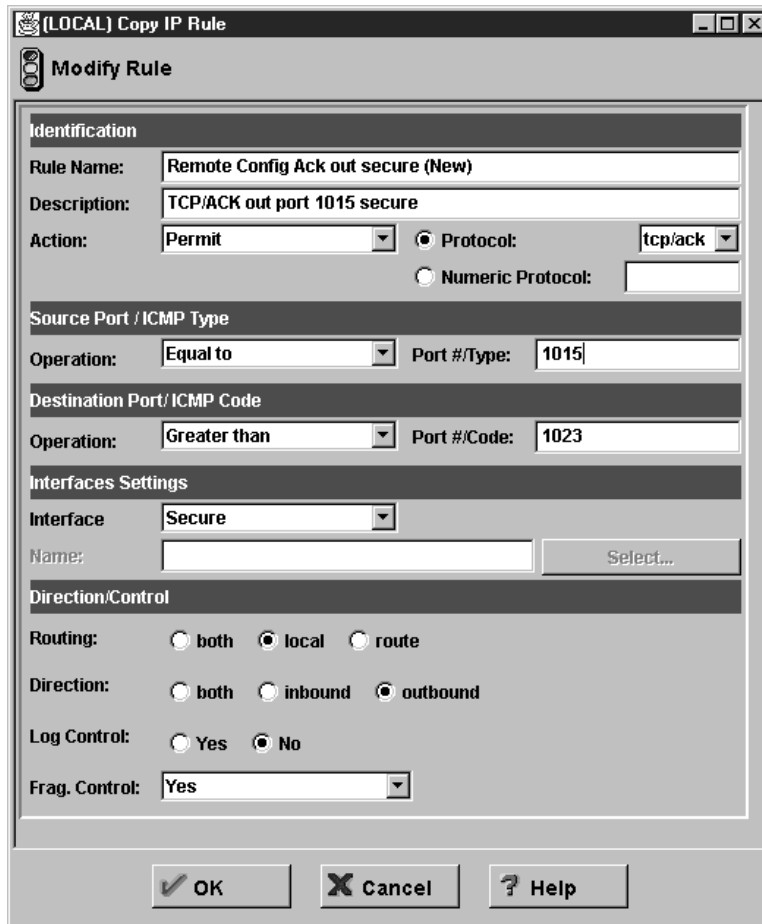


Figure 223. Creating (by Copying) a Remote Configuration Ack Out Rule

4. After creating the new rules, you have to define the new service. Open the **Services** document in the Connection Templates folder of the navigation tree.
5. Click on **<NEW>** to add a new service. Give it the parameters shown in Figure 224 on page 243. Use the new rules you've defined.

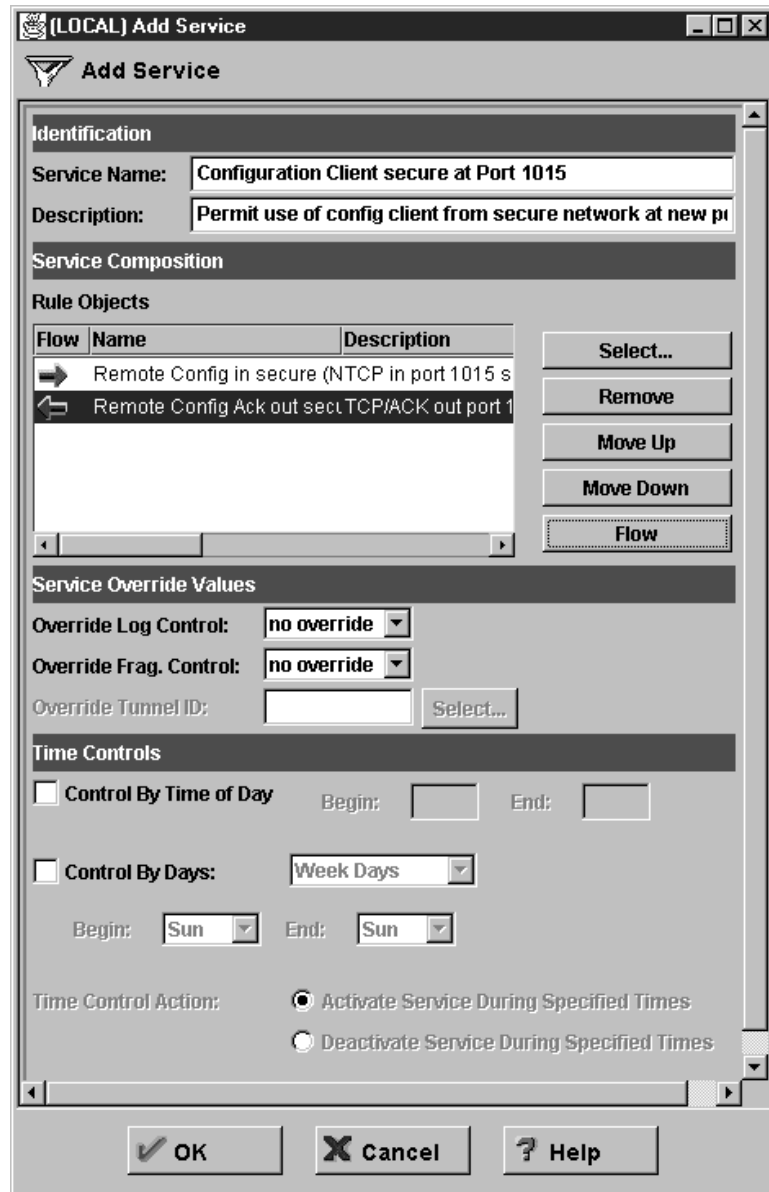


Figure 224. Creating the Service for the New Port

6. After adding the service, go back to the Navigation Tree and open the **Connection Setup** document.
7. Click on **<NEW>** to add a new connection. Use the new service you've defined, as shown in Figure 225 on page 244.

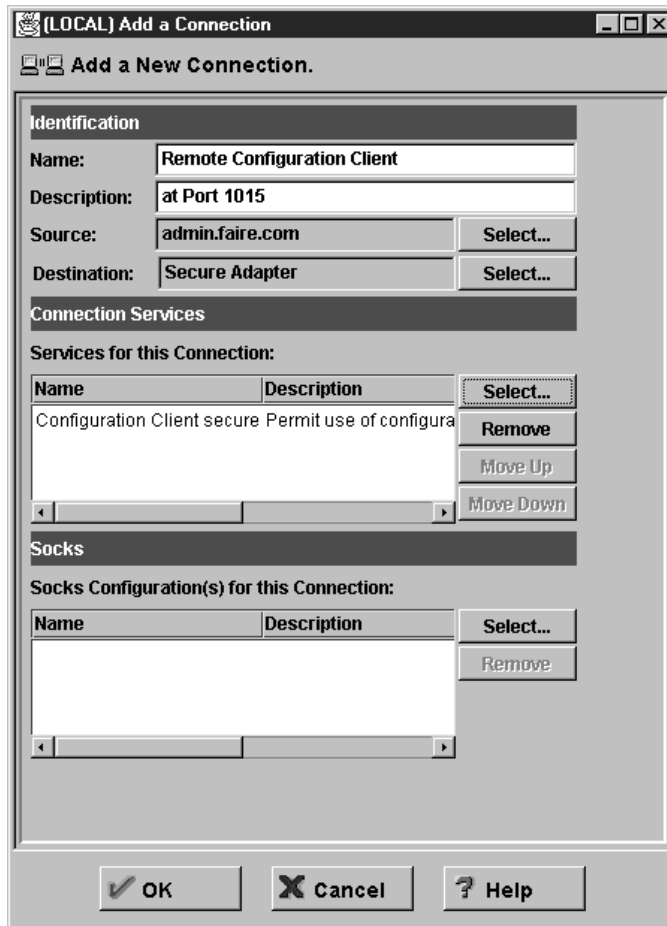


Figure 225. Creating the Connection for the New Port

8. After creating the connection, regenerate the rules.

10.5.2 Logging in Using the New Port Numbers

You'll have to use the new port number every time you log on to the firewall. This applies whether you're logging on locally or from a remote client.

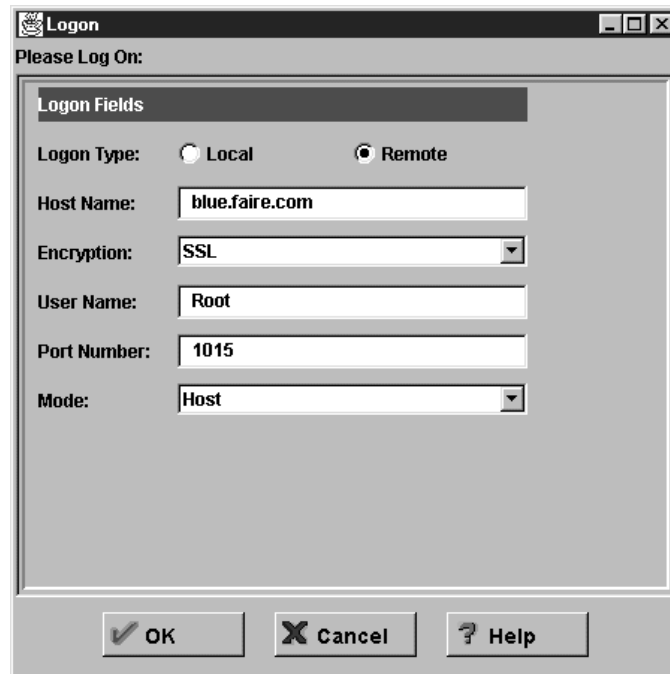


Figure 226. Logging On at the New Port Number

10.6 From Here On...

In this chapter, we've shown you how to set the IBM Firewall to accept connections from remote clients, and how to set up the administrator accounts who will use them. We've also discussed how to secure the remote client connections using SSL and specific filter rules and connections. Last of all, we showed you how to change the port number of the Configuration Server, and how to create filter rules and connections for the new port number.

Some suggested chapters to read:

- For a more detailed discussion of packet filter rules, see Chapter 8, "IBM Firewall Rule Base" on page 143.
- To set up a SecurID server and configure the firewall to use it for authentication, read 21.5, "SecurID Authentication" on page 563.

Chapter 11. Domain Name Service (DNS)

The Domain Name System is an important part of the Internet today. Its main function is to provide information about Internet hosts. It resolves the IP address by using a name or vice versa.

This chapter contains three sections. The list below explains briefly the contents of all sections.

- In the first section we explain how DNS works. The objective of this section is not for explaining DNS in detail. It describes important concepts that are closely related with firewall configuration. Those who already know DNS can skip it and jump directly to the second sub-chapter.
- The second sub-chapter talks about some of the attacks that a company might have when using DNS.
- The third section discusses the mechanism that IBM eNetwork Firewall for Windows NT uses to address the issues described in second section.
- In section four, we give an example of a DNS configuration using firewall and the testing of the DNS configuration.

Now, let's start with DNS.

11.1 How Does DNS Work?

The Domain Name System (DNS) is a hierarchical, client/server-based distributed database management system. The purpose of the DNS database is to resolve resource names to IP addresses or vice versa. In the DNS, the clients are called resolvers and the servers are called name servers. DNS uses either UDP (port 53) or TCP (port 53). Resolvers first send UDP queries to servers for increased performance and resort to TCP only if truncation of the returned data occurs. Servers always use TCP for replicating database information.

The DNS name space is in the form of a logical tree structure. Each branch point (or node) in the tree is given a name up to 63 characters long. The root of the tree is given the special reserved name of "" (null). Each node is a partition of the name space called a *domain*. A domain is a branch of the tree and can occur at any point in the tree structure.

Domains can be further partitioned at node points within the domain into *subdomains* for the purpose of administration or load balancing. The *domain name* identifies the domain's position in the DNS hierarchy. The *full domain name* identifies the domain relative to the root. Domain names and full domain names are created by combining the names of the nodes from the designated domain node back to the root and separating each node with a period (.). The global picture of DNS name space is shown in Figure 227 on page 248.

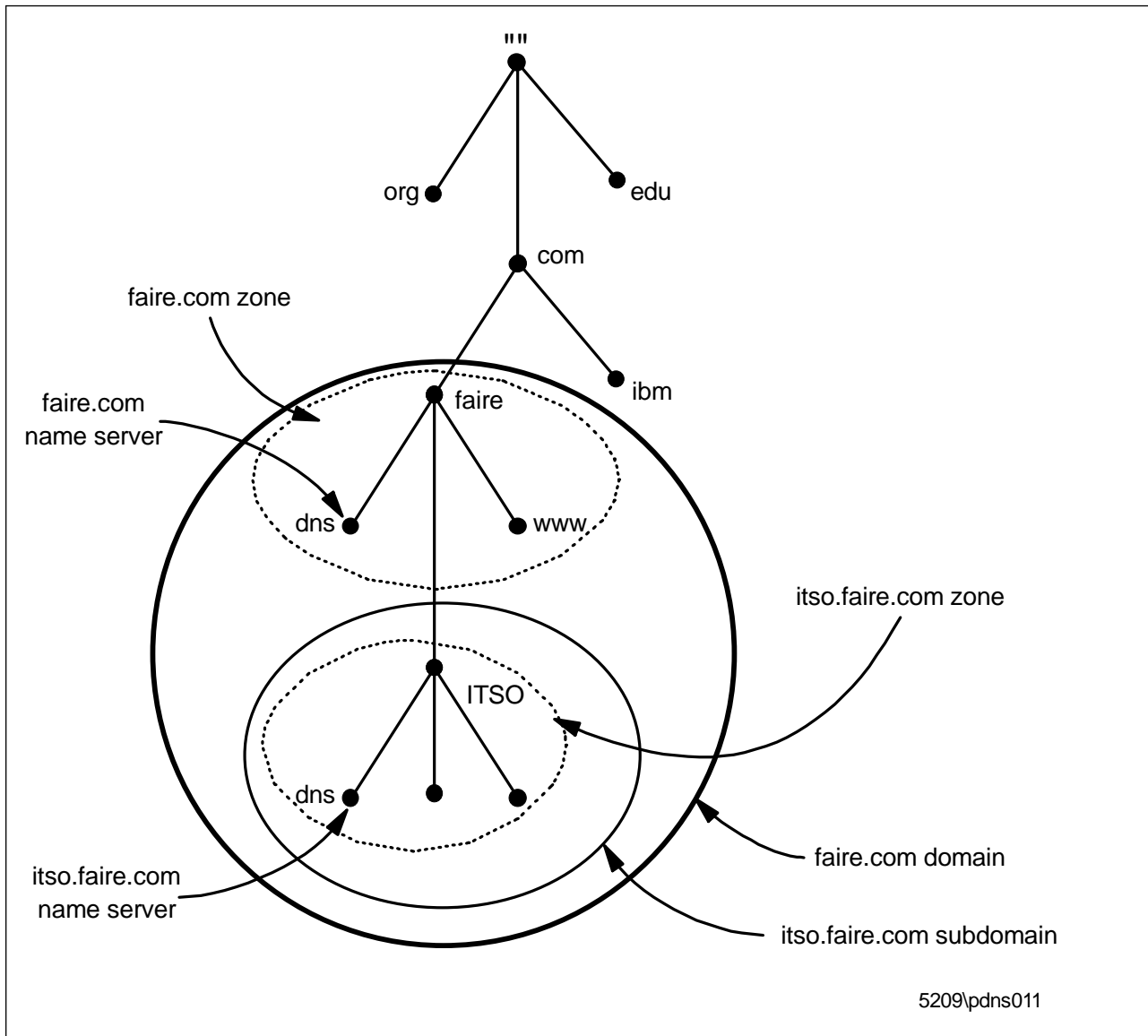


Figure 227. Example of DNS Name Space

Host names are added to the beginning of the domain name and are often referred to by their fully qualified domain name. For example, a host named www in the securez.com domain would have the fully qualified domain name of www.securez.com. A host name is a name assigned to a computer by an administrator to identify a TCP/IP host.

The programs that store information about the domain name space are called *name servers*. Name servers generally have completed information about some part of the domain name space, called a *zone*. As we can see in Figure 227 faire.com has two zones. One zone is for faire.com and the other is for itso.faire.com. The name server is then said to have *authority* for that zone. That's why we sometimes receive a non-authoritative answers since the response comes from the zone outside our name server's authority. In Figure 227 we can see that two different zones have two different name servers. We use this approach since we use the firewall to divide the domain and subdomain. The faire.com domain is for a non-secure network and itso.faire.com is for a secure network.

A *Primary Name Server* is a name server that gets the data for its zones from locally stored and maintained files. Changes to a zone, such as adding domains or hosts, are done by changing files at the primary name servers. A *Secondary Name Server* gets the data for its zones from another name server, (either a primary name server or another secondary name server) for that zone across the network. The process of obtaining this zone information (that is, the database file) across the network is referred to as a *zone transfer*. Zone transfers use TCP port 53.

On a globally connected network like the Internet, DNS queries outside a local zone may require interaction with DNS name servers across wide area network links outside of the company. What is needed is a way to designate specific name servers as being responsible for WAN-based DNS traffic. To address this issue DNS allows for the concept of *forwarders*. Specific DNS name servers are selected to be forwarders and usually forwarders are to carry out the wide area communications across the Internet. In nonexclusive mode, when a name server receives a DNS query that it cannot resolve through its zone files, it passes the query to one of the designated forwarders. If the forwarder is unable to resolve the query, the name server that received the original query attempts to resolve the query on its own. In an exclusive mode, name servers rely completely on the name-resolving ability of the forwarders. Name servers using forwarders in an exclusive mode are known as *slaves*. Slaves make no attempt to resolve the query on their own if the forwarder is unable to satisfy the request.

Although all DNS name servers cache queries that they have resolved, *Caching-only Servers* are name servers whose only job is to perform queries, cache the answers, and return the results. Caching-only servers are not authoritative for any domains and contain only information that they have cached while resolving queries.

Another major functionality missing from the resolution process as explained so far is how IP addresses get mapped back to names. Address-to-name mapping is used to produce output that is easier for humans to read and interpret (in log files, for instance). To prevent an exhaustive search of all domains for an inverse query, a special domain IN-ADDR.ARPA was created. Nodes in the IN-ADDR.ARPA domain are named after the numbers in the dotted decimal representation of IP addresses. But since IP addresses get more specific from left to right and domain names get less specific from right to left, the order of IP address octets must be reversed when building the IN-ADDR.ARPA domain. The domain of IN-ADDR.ARPA is shown in Figure 228 on page 250.

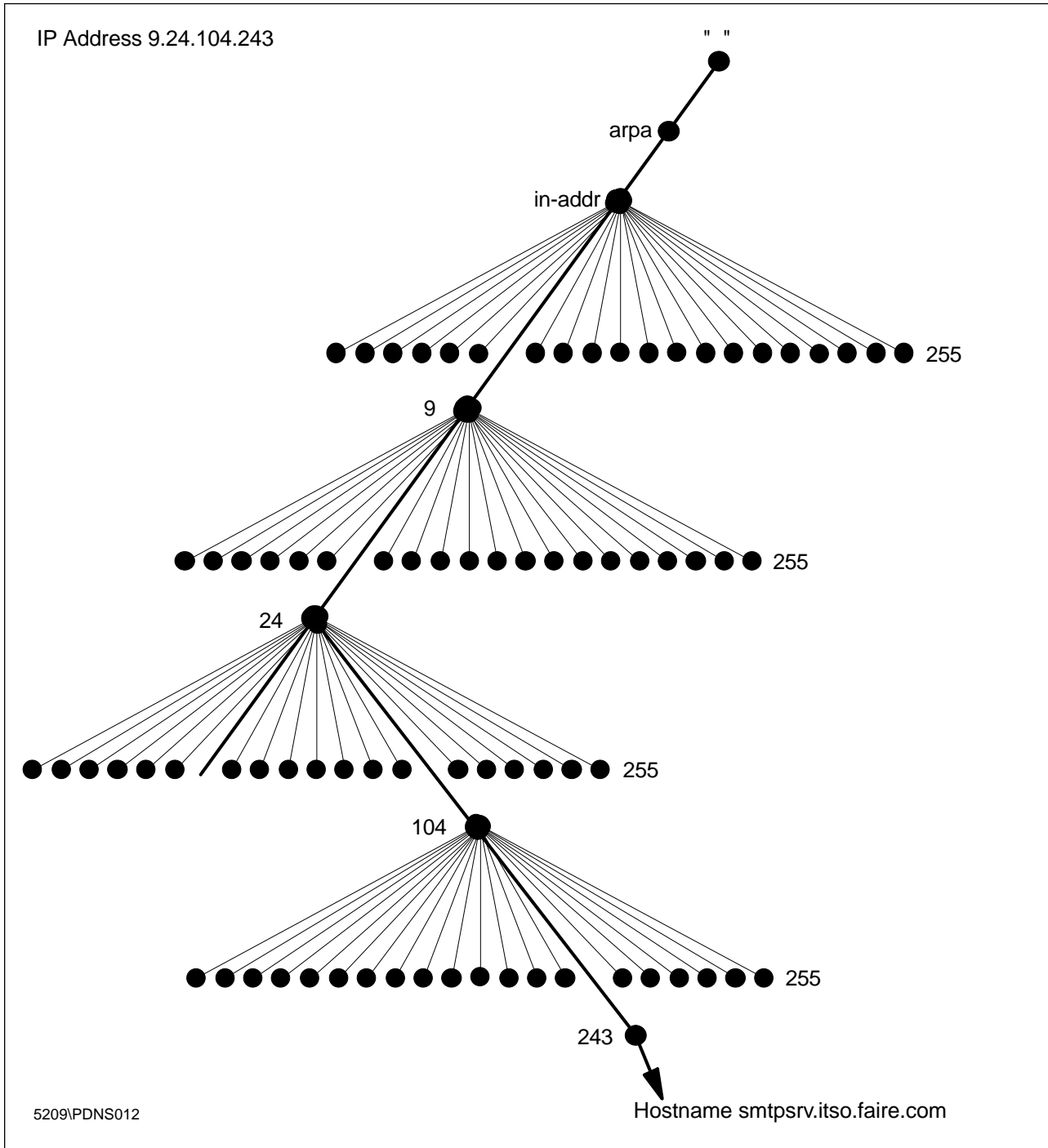


Figure 228. The IN-ADDR.ARPA Domain

Once the IN-ADDR.ARPA domain is built, special resource records called PTR (pointer) records are added to associate the IP address to the corresponding host names. To find a host name for the IP address 9.24.104.243, the resolver would query the DNS server for a PTR record for 243.104.24.9.IN-ADDR.ARPA.

A domain name defines a node within the DNS name space. Each node is comprised of a set of *resource records* which make up the resource information associated with the domain. The resource records are stored in files which are

referred to as *DNS Database Files* or simply *DB Files*. Those DB files and their functions are shown below.

Name	Description
DB.DOMAIN	This is a file that maps host names to IP addresses. For example, if your domain is <i>faire.com</i> , then this file will be called <i>db.faire</i> .
DB.ADDR	This is a file that maps IP addresses to host names. For example, if your network is the class A network address <i>9.24.104.0</i> , then this file will be called <i>db.9.24.104</i> .
DB.CACHE	This file is essentially the same on all name servers and must be present. This file contains the names and addresses for the name servers that maintain the root domain.
DB.127.0.0	This file is essentially the same on all name server and must be present. This file is used to resolve queries to the loop-back address.
boot file	This file ties all the db files together. This file is needed by a name server as a startup file.

For detailed information about DNS, you can read a good book from O'Reilly & Associates, Inc. entitled *DNS and BIND* and written by Paul Albitz & Cricket Liu.

11.2 Some Attacks on DNS

There are some common attacks on DNS. We will give a brief description of two of those attacks and hopefully it will provide a good picture of the threats that may happen in a DNS environment.

11.2.1 Cache Poisoning

Cache poisoning occurs when malicious or misleading data received from a remote name server is saved (cached) by another name server. This *bad* data is then made available to programs that request the cache data through the client interface. The cache poisoning technique is being used to adverse effect the mapping between host names and IP addresses. Once this mapping has been changed, any information sent between hosts on a network may be subjected to inspection, capture, or corruption.

The common way to poison the cache is to predict the sequence of query IDs when the name server is resolving recursive queries. If the DNS server uses a predictable sequence, then it is easy to attack the server. For those who use the Microsoft DNS service, the *dnsfix* is provided by Microsoft technical support. It can be found at:

<ftp://ftp.microsoft.com/bussys/winnt/winnt-public/fixes/usa/nt40/hotfixes-postSP3/dns-fix/>

With that fix, instead of using a predictable sequence, it will use a random sequence to reduce the effectiveness of this cache poisoning attack. It is the reason why we have to apply the fix before installing IBM Firewall for NT. See 4.4, "Post-Installation Procedures" on page 47 for details. For those who use other name servers, you can possibly find DNS fixes for your system at the following ftp site:

ftp://ftp.cert.org/pub/cert_advisories/CA-97.22.bind

11.2.2 UDP Denial-of-Service Attack

When a connection is established between two UDP services, each of which produces output, these two services can produce a very high number of packets that can lead to a denial of service on the machine(s) where the services are offered. Anyone with network connectivity can launch an attack; no account access is needed. For example, by connecting a host's charge service to the echo service on the same or another machine, all affected machines may be effectively taken out of service because of the excessively high number of packets produced. In addition, if two or more hosts are so connected, the intervening network may also become congested and deny service to all hosts whose traffic traverses that network.

For those who use the Microsoft DNS, this issue has also been resolved by the same fix as stated in 11.2.1, "Cache Poisoning" on page 251. And for those who use other name server you can also find DNS fixes for your system at the following ftp site:

ftp://ftp.cert.org/pub/cert_advisories/CA-96.02.UDP_service_denial

11.3 Implementation of DNS in IBM Firewall

After we discussing some of the attacks on the DNS environment, we will now look at how IBM Firewall for NT provides a secured DNS implementation. Generally, what the firewall is supposed to do is divided into two aspects.

11.3.1 The Firewall Hides the Information about Internal Hosts

When somebody tries to find name information of the internal network, the DNS resolution will stop on the external DNS. The only information from the external DNS that specify the firewall is the MX records that will forward all incoming mails. The mail forwarding is discussed in Chapter 12, "Mail Handling" on page 269. With this configuration, there is no way for the intruders to find names information from the internal DNS. The external DNS only contains information about hosts in the DMZ.

11.3.2 The Firewall Permits DNS Resolution of External Hosts

This involves queries for external names sent to the internal name server. Rather than talking to the real server (which we can't allow it even though it is possible to open port 53 for both directions), the internal DNS forwards the request to the gateway. It can be done by adding the *forwarders* directive in the DNS boot file. The firewall will safely send the query to the external name server. It is safe since the DNS in the firewall is a caching-only name server. It means that the firewall is *not* responsible for a zone. What it does is only perform a query, cache the answer and return the results.

Now, how does the application in the firewall, for example proxy server, resolve the external names? It will ask the internal DNS first. Of course, the internal DNS doesn't know the answer unless it's in the cache. Then, the internal DNS follows the same procedure, forward to the firewall, the firewall asks the external DNS and so on. The response from the external DNS will be sent to the firewall and then to

the internal DNS; lastly the internal DNS sends the response to the application in the firewall.

Figure 229 shows the flow of name resolution using firewall.

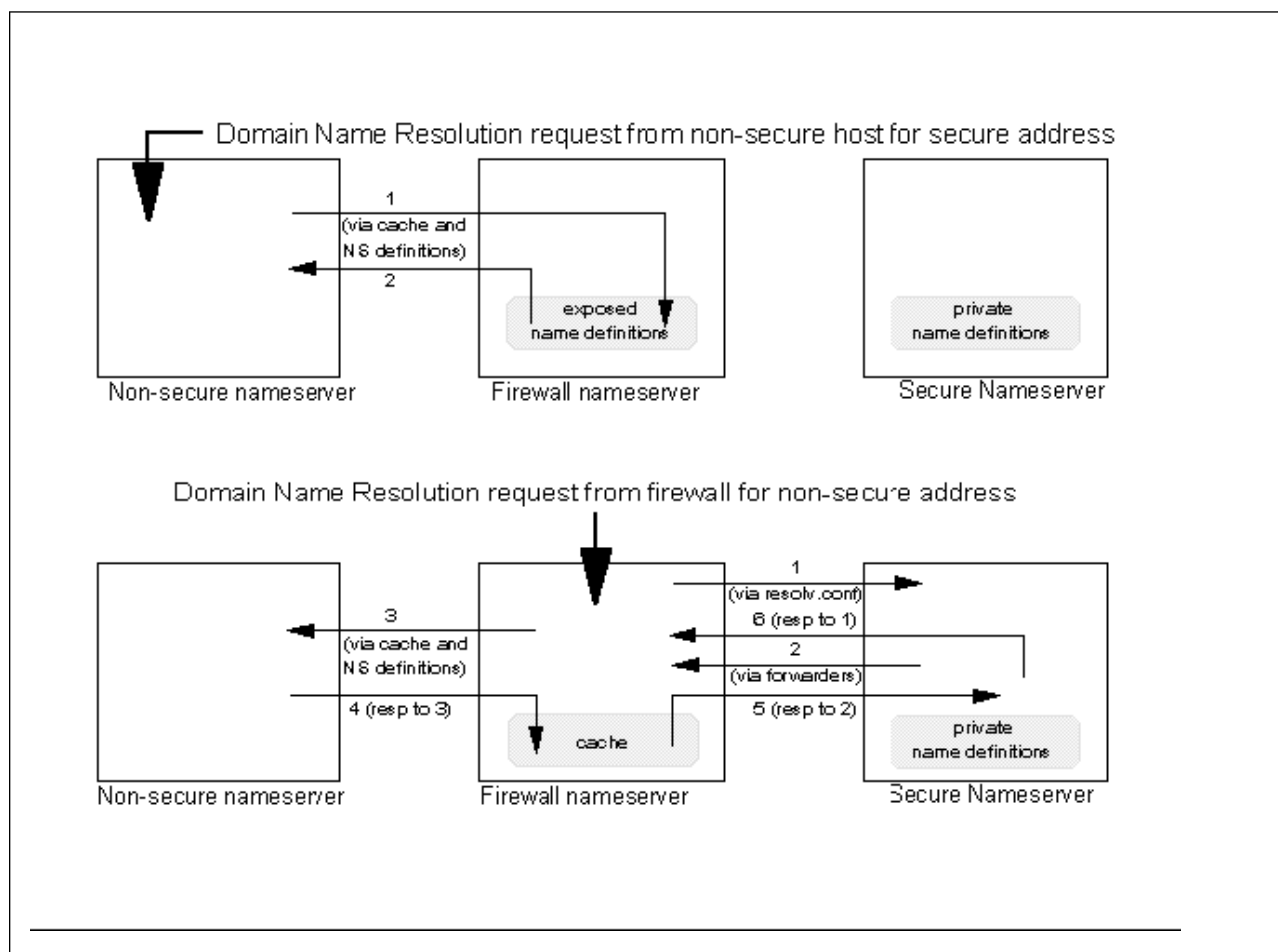


Figure 229. Name Resolution Flows

To set up the DNS environment, the first thing that we have to do is define the domain name for the internal and the external network. Those possible configurations of domain names are already discussed in 3.6, "Naming Services" on page 36. Also you have to decide whether to use separate external DNS or not. Please refer to 3.6, "Naming Services" on page 36 for discussion about the external mail server.

Now, you are ready to set up the internal and external DNS that are described in the examples below in 11.4.1, "Configuration of the Internal DNS" on page 255 and 11.4.2, "Configuration of the External DNS" on page 259 respectively. The differences between the separate external DNS and external DNS in the firewall is not too far apart. The consequence is that the firewall now is not only the caching-only name server but it is now maintaining a zone. But still it only contains the DMZ hosts information.

The last thing is to set up the DNS in the firewall, which is very simple by using GUI and testing the configuration. These are described in 11.4.3, "Configuration of

DNS in Firewall” on page 263 and 11.4.4, “Testing The DNS Configuration” on page 268 respectively.

11.4 Examples of DNS Configuration

Now, let’s take a look at the configuration of DNS we’ve set up. As we explained before, we use internal and external DNS. Below is the diagram of our DNS environment.

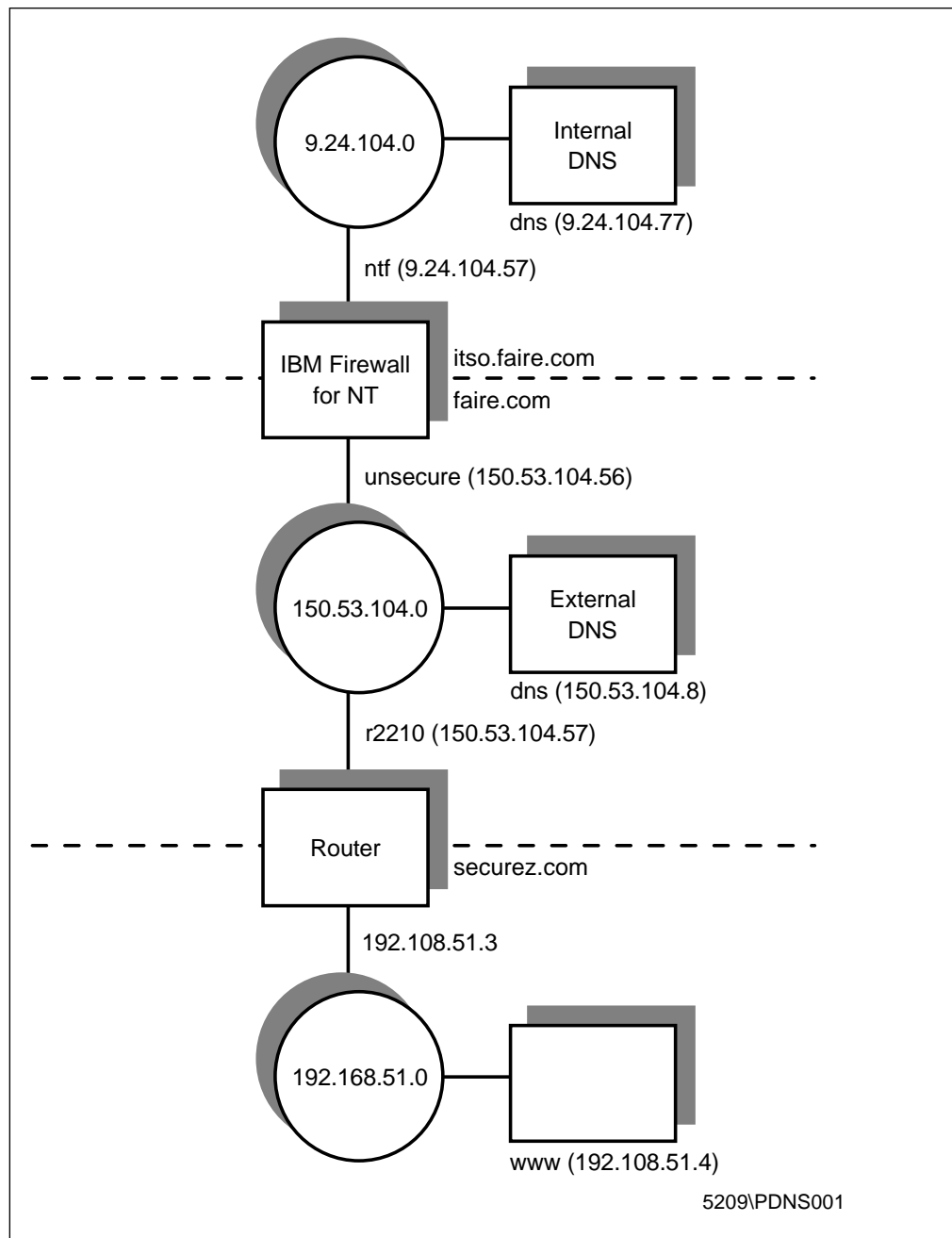


Figure 230. Example of DNS Configuration

We use NT Server 4.0 for all machines. We use Microsoft DNS Service for both internal and external DNS as well as the DNS in the firewall.

11.4.1 Configuration of the Internal DNS

The internal DNS has no direct access to the Internet, so it cannot access the root nameservers. In order to resolve non-secure host names, we have to configure the Internal DNS to point to the secure interface of the firewall. Since we use Microsoft DNS, we add the forwarder parameter using the GUI and opening the **DNS Manager** window from **Administrative Tools**. Open the **Properties** of the server as shown in Figure 231.

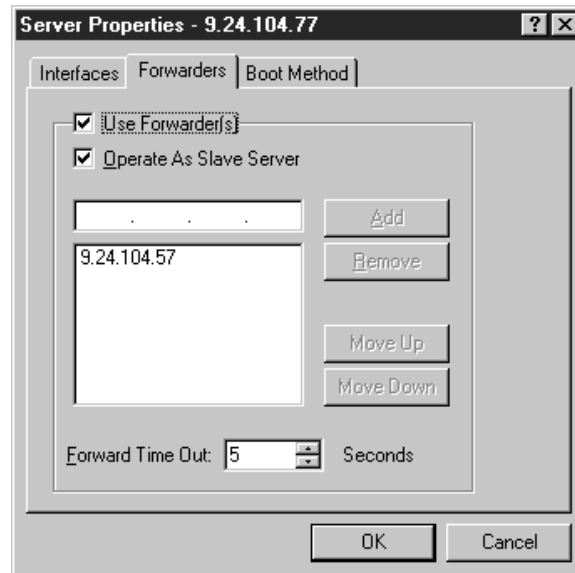


Figure 231. Adding Forwarder Using Microsoft DNS Manager

Or, if you use other DNS you should add the following line in the name server boot file:

```
forwarders 9.24.104.57
```

The internal name server will forward all queries about external Internet hosts. What it does is send the query to a forwarder and wait a short period of time for an answer before resuming normal operation. The internal name server will wait for a while to see whether the firewall has the answer in its cache file. If it does, then the firewall will instantly response the query.

As you can see from Figure 231 we also check on the **Operate As Slave Server**. This means that the internal name server will not contact other server to find out information if the forwarders don't give an answer. It relies *completely* on the forwarders. For details see 11.1, "How Does DNS Work?" on page 247. You can add the slave directive in your DNS boot file manually as shown below:

```
forwarders 9.24.104.57 9.24.104.57 9.24.104.57
slave
```

Please note here that since we use the slave directive, which means that the internal name server relies completely on the name-resolving ability of the forwarder, we put the forwarder multiple times to increase the overall length of time that the slave name server will wait for an answer from the forwarder. In the case of Microsoft DNS, we can change the timeout parameter from the registry instead, as shown in Figure 232 on page 256.

If you use Microsoft DNS service, after configuring the DNS using DNS manager, Windows NT will put the information of DNS boot file in the registry. The location is in:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\DNS\Parameters

Figure 232 shows the Microsoft DNS boot information.

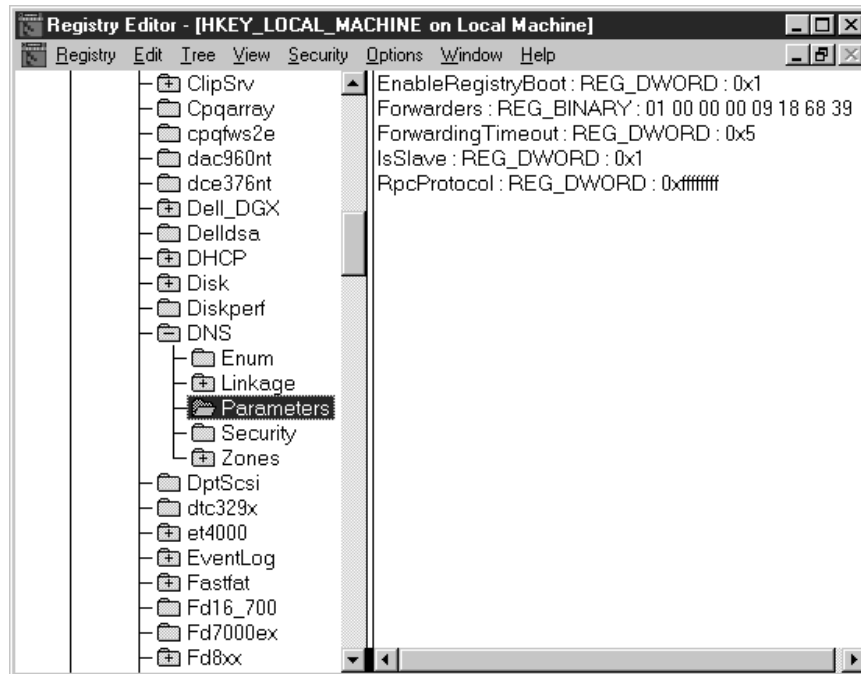


Figure 232. Microsoft DNS Boot File Information in NT Registry

If you want to use the usual boot file, you have to change the value of the EnableRegistryBoot variable to 0. The registry contains five variables, which are explained below:

- | | |
|---------------------------|---|
| EnableRegistryBoot | 0x1 means that Windows NT will use the registry for DNS boot file. Windows NT will use the usual boot file if we change the value to 0x0. |
| Forwarders | Contains the IP address of the forwarder in hexadecimal format. |
| ForwardingTimeout | Contains the time in seconds to wait for the forwarder to resolve names. |
| IsSlave | 0x1 means that the internal name server is a slave name server. |
| RpcProtocol | Means Windows NT uses RPC protocol for DNS. |

The remaining configuration files of the internal DNS are conventional. In the case of Windows NT, those files are placed in directory:

c:\winnt\system32\dns

We show all configuration files for completeness.

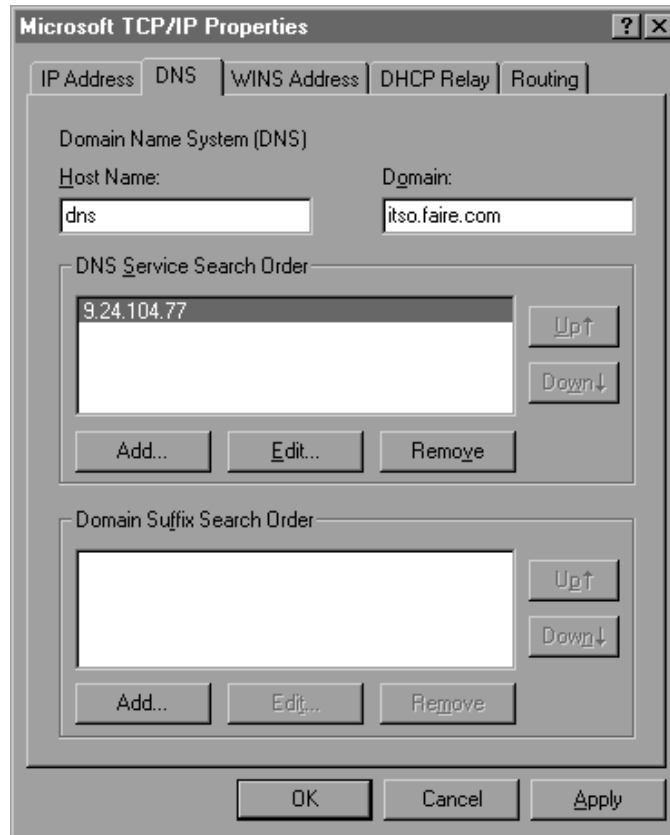


Figure 233. Windows NT Resolver Configuration on the Internal DNS

Please refer to the *Windows NT User's Guide* for more information on how to configure Microsoft's DNS.

```

;
; Database file itso.faire.com.dns for itso.faire.com zone.
;   Zone version: 41
;
@           IN      SOA      dns.itso.faire.com.  Administrator.itso.faire.com
          (
          4          ; serial number
          3600       ; refresh
          600        ; retry
          86400      ; expire
          3600      ) ; minimum TTL

;
; Zone NS records
;
@           IN      NS       dns

;
; Zone records
;
andre      IN      A         9.24.104.243
dns        IN      A         9.24.104.77
dominic    IN      A         9.24.104.113
ntf        IN      A         9.24.104.57
sven       IN      A         9.24.104.70
belloni    IN      A         9.24.104.39
smtpsrv    IN      CNAME     andre.itso.faire.com.

```

Figure 234. c:\winnt\system32\dns\itso.faire.com.dns File on the Internal DNS

The above file contains all information that is managed by the itso.faire.com zone.

```

;
; Database file 104.24.9.in-addr.arpa.dns for 104.24.9.in-addr.arpa zone
;   Zone version: 41
;
;
@           IN      SOA      dns.itso.faire.com.  Administrator.itso.faire.com (
          4          ; serial number
          3600       ; refresh
          600        ; retry
          86400      ; expire
          3600      ) ; minimum TTL

;
; Zone NS records
;
@           IN      NS       dns.itso.faire.com.

;
; Zone records
;
113        IN      PTR     dominic.faire.com.
243        IN      PTR     andre.faire.com.
57         IN      PTR     ntf.faire.com.
70         IN      PTR     sven.faire.com.
77         IN      PTR     dns.faire.com.
39         IN      PTR     belloni.faire.com.

```

Figure 235. c:\winnt\system32\dns\104.24.9.in-addr.arpa.dns File on the Internal DNS

The above file contains all reverse addresses of all hostnames in the itso.faire.com domain. It is managed by an independent zone called 104.24.9.in-addr.arpa zone.

The DNS Server is now booting from the registry. New zones or changes to zone information made through the DNS Manager, require the DNS server to boot from information stored in the registry. Further changes to zone configuration MUST be made through the DNS Manager as the boot file is no longer being read. The previous boot file has been moved to the backup directory.

If you must return to using a boot file, use the registry editor to open HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\DNS\Parameters and delete the value EnableRegistryBoot. Then replace your desired boot file in the system32\dns directory. Note, however, that ALL changes to zone information, including new zones, made through the DNS Manager will be lost.

Figure 236. c:\winnt\system32\dns\boot File on the Internal DNS

Figure 236 shows the way Microsoft DNS service running the boot file as discussed earlier.

11.4.2 Configuration of the External DNS

The external DNS consists of information about hosts in the DMZ. The only record that points to the non-secure interface of the firewall is an MX record. This is needed to resolve incoming mail. Another MX record is added in the securez.com zone information to resolve external SMTP server which is located in the securez.com domain. The next seven figures show the configuration of the external DNS.

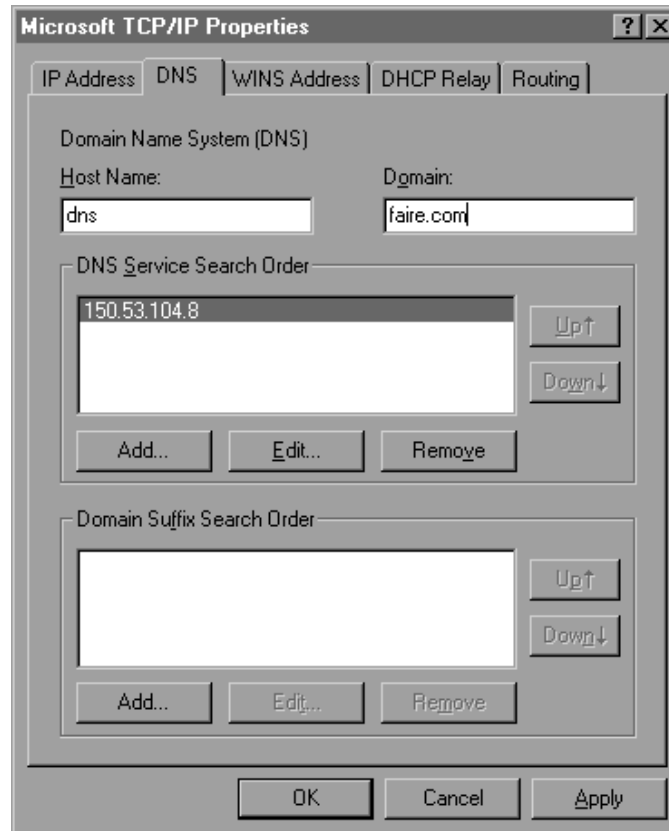


Figure 237. Windows NT Resolver Configuration in the External DNS

```

;
; Database file faire.com.dns for faire.com zone.
;   Zone version: 91
;
@           IN      SOA    dns.faire.com.  Administrator.faire.com.
(
          9          ; serial number
          3600       ; refresh
          600        ; retry
          86400      ; expire
          3600       ) ; minimum TTL

;
; Zone NS records
;
@           IN      NS     dns

;
; Zone records
;
@           IN      MX     10    nonsecure
r2210      IN      A      150.53.104.57
nonsecure  IN      A      150.53.104.56
dns        IN      A      150.53.104.8
www        IN      CNAME   dns

```

Figure 238. c:\winnt\system32\dns\faire.com.dns File on the External DNS

The file above contains information about hostnames in faire.com zone. An MX record tells the external name server to send all mail destined to faire.com to nonsecure.faire.com, which is the non-secure interface of the firewall.

```

;
; Database file 104.53.150.in-addr.arpa.dns for 104.53.150.in-addr.arpa.zone.
;   Zone version: 61
;
@           IN      SOA    dns.faire.com.  Administrator.faire.com.
(
          6          ; serial number
          3600       ; refresh
          600        ; retry
          86400      ; expire
          3600       ) ; minimum TTL

;
; Zone NS records
;
@           IN      NS     dns.faire.com.

;
; Zone records
;
56          IN      PTR    nonsecure.faire.com.
57          IN      PTR    r2210.faire.com.
8           IN      PTR    dns.faire.com.

```

Figure 239. c:\winnt\system32\dns\104.53.150.in-addr.arpa.dns File on the External DNS

The file above contains reverse information about hostnames in the faire.com domain. The information is managed by another zone called 104.53.150.in-addr.arpa..

The DNS Server is now booting from the registry. New zones or changes to zone information made through the DNS Manager, require the DNS server to boot from information stored in the registry. Further changes to zone configuration MUST be made through the DNS Manager as the boot file is no longer being read. The previous boot file has been moved to the backup directory.

If you must return to using a boot file, use the registry editor to open HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\DNS\Parameters and delete the value EnableRegistryBoot. Then replace your desired boot file in the system32\dns directory. Note, however, that ALL changes to zone information, including new zones, made through the DNS Manager will be lost.

Figure 240. c:\winnt\system32\dns\boot File on the External DNS

The boot file above contains the same information with the boot file of internal name server. It means that Windows NT uses Registry to read the DNS boot file.

```

;
; Cache file:
;
.                2163095040    IN    NS    A.ROOT-SERVERS.NET.
A.ROOT-SERVERS.NET. 2163095040    IN    A     198.41.0.4
.                2163095040    IN    NS    B.ROOT-SERVERS.NET.
B.ROOT-SERVERS.NET. 2163095040    IN    A     128.9.0.107
.                2163095040    IN    NS    C.ROOT-SERVERS.NET.
C.ROOT-SERVERS.NET. 2163095040    IN    A     192.33.4.12
.                2163095040    IN    NS    D.ROOT-SERVERS.NET.
D.ROOT-SERVERS.NET. 2163095040    IN    A     128.8.10.90
.                2163095040    IN    NS    E.ROOT-SERVERS.NET.
E.ROOT-SERVERS.NET. 2163095040    IN    A     192.203.230.10
.                2163095040    IN    NS    F.ROOT-SERVERS.NET.
F.ROOT-SERVERS.NET. 2163095040    IN    A     39.13.229.241
.                2163095040    IN    NS    G.ROOT-SERVERS.NET.
G.ROOT-SERVERS.NET. 2163095040    IN    A     192.112.36.4
.                2163095040    IN    NS    H.ROOT-SERVERS.NET.
H.ROOT-SERVERS.NET. 2163095040    IN    A     128.63.2.53
.                2163095040    IN    NS    I.ROOT-SERVERS.NET.
I.ROOT-SERVERS.NET. 2163095040    IN    A     192.36.148.17

```

Figure 241. c:\winnt\system32\dns\CACHE.DNS File on the External DNS

The file above is created by the Microsoft DNS and contains the list of name servers for the root domain. This information must be updated from the Internet host ftp.rs.internic.net (198.41.0.5). Use anonymous FTP to retrieve the file *named.root* from the domain subdirectory.

As you can see in Figure 230 on page 254, we have another domain to represent the Internet called securez.com. For simplicity, we put securez.com zone information in the external name server. The file above is the information of all hostnames in securez.com domain. The two configuration files are shown in Figure 242 on page 262 and Figure 243 on page 262.

```

;
; Database file securez.com.dns for securez.com zone.
;   Zone version: 61
;
@           IN      SOA      smtpsrv.faire.com.  Administrator.faire.com.
(
        6          ; serial number
       3600        ; refresh
        600        ; retry
       86400       ; expire
       3600       ) ; minimum TTL

;
; Zone NS records
;
@           IN      NS       smtpsrv.faire.com.

;
; Zone records
;
www         IN      CNAME    smtpsrv
smtpsrv     IN      A        192.168.51.4

```

Figure 242. *c:\winnt\system32\dns\securez.com.dns File on the External DNS*

The file above contains host name information about securez.com

```

.
;
; Database file 51.168.192.in-addr.arpa.dns for 51.168.192.in-addr.a
;   Zone version: 21
;
@           IN      SOA      smtpsrv.faire.com.  Administrator
(
        2          ; serial number
       3600        ; refresh
        600        ; retry
       86400       ; expire
       3600       ) ; minimum TTL

;
; Zone NS records
;
@           IN      NS       smtpsrv.faire.com.

;
; Zone records
;
4          IN      PTR      smtpsrv.securez.com.

```

Figure 243. *c:\winnt\system32\dns\51.168.192.in-addr.arpa.dns File on the External DNS*

The file above contains reverse information of the securez.com domain. And it is managed by another zone called 51.168.192.in-addr.arpa.

11.4.3 Configuration of DNS in Firewall

After we successfully configure the internal and external DNS, it is very easy to set up the DNS function in the firewall. Log on to the firewall using configuration client. Get to the first screen and double-click on **System Administration** to pull the menu down.

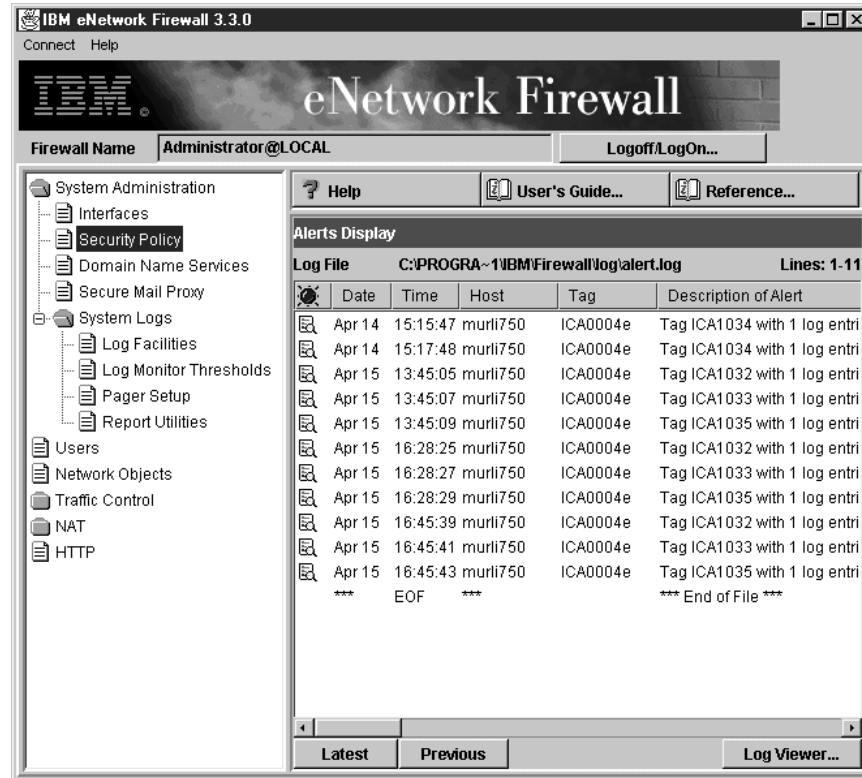


Figure 244. Security Policy in the Firewall

Double-click on **Security Policy** to open the Security Policy screen.

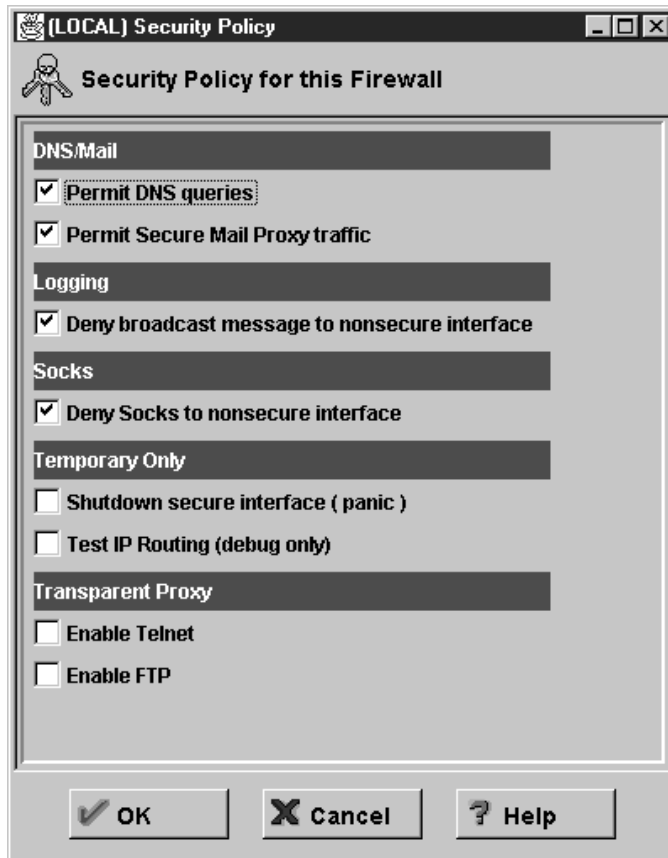


Figure 245. Open DNS Traffic in the Firewall

Check on the **Permit DNS queries** to enable DNS queries. Click on **OK** to save the configuration and get back to the main menu. Basically, when we check **Permit DNS queries** on and click on **OK** the firewall will create connections from pre-defined services. See Chapter 9, "Examples of Rules for Specific Services" on page 161 for more information. The pre-defined rules are shown below:

```

permit 0 0 0 0 udp gt 1023 eq 53 both local both
permit 0 0 0 0 udp eq 53 gt 1023 both local both
permit 0 0 0 0 udp eq 53 eq 53 both local both

```

You should also allow TCP traffic, not just UDP. DNS uses TCP for two situations:

- to allow DNS zone transfers; a zone transfer occurs when a secondary name server requests a primary name server to send all it knows about a zone.
- DNS long queries.

In order to create connections to allow TCP traffic, you should use the pre-defined service called DNS transfers. The connection rules should be:

```

permit 0 0 0 0 tcp eq 53 eq 53 both local both
permit 0 0 0 0 tcp gt 1023 eq 53 both local both
permit 0 0 0 0 tcp/ack eq 53 gt 1023 both local both

```

The IBM eNetwork Firewall for AIX automatically creates the first rule when you check **Permit DNS queries** in the Security Panel to allow DNS long queries. The IBM eNetwork Firewall for NT will also do it in a future release.

The above rules mean that all hosts in the secure network can access the secure interface of the firewall for DNS. Likewise, all hosts in the non-secure network are able to access the non-secure interface of the firewall for DNS.

You could limit the access to the firewall DNS from the secure network to only the internal DNS name server, but you still need to keep the non-secure interface able to connect to The World:

```
#DNS from Internal Name Server to Secure Interface (1)
permit 9.24.104.77 0xffffffff 9.24.104.57 0xffffffff udp gt 1023 eq 53 secure local inbound
permit 9.24.104.57 0xffffffff 9.24.104.77 0xffffffff udp eq 53 gt 1023 secure local outbound

#DNS from Secure Interface to Internal Name Server (2)
permit 9.24.104.57 0xffffffff 9.24.104.77 0xffffffff udp gt 1023 eq 53 secure local outbound
permit 9.24.104.77 0xffffffff 9.24.104.57 0xffffffff udp eq 53 gt 1023 secure local inbound

#DNS from Internal Name Server to Secure Interface (1)
permit 9.24.104.77 0xffffffff 9.24.104.57 0xffffffff udp eq 53 eq 53 secure local inbound

#DNS from Secure Interface to Internal Name Server (2)
permit 9.24.104.57 0xffffffff 9.24.104.77 0xffffffff udp eq 53 eq 53 secure local outbound
#DNS from Non-secure Interface to The World (1)
permit 150.53.104.56 0xffffffff 0 0 udp eq 53 eq 53 non-secure local outbound

#DNS from The World to Non-secure Interface (2)
permit 0 0 150.53.104.56 0xffffffff udp eq 53 eq 53 non-secure local inbound
```

The new rules above were set for our DNS configuration, where the IP address of the internal DNS is 9.24.104.77, the IP address of the secure interface of the firewall is 9.24.104.57. and the IP address of the non-secure interface of the firewall is 150.53.104.56.

You could also restrict the DNS TCP traffic for DNS zone transfers in a similar fashion.

Now, we will configure DNS in the firewall.

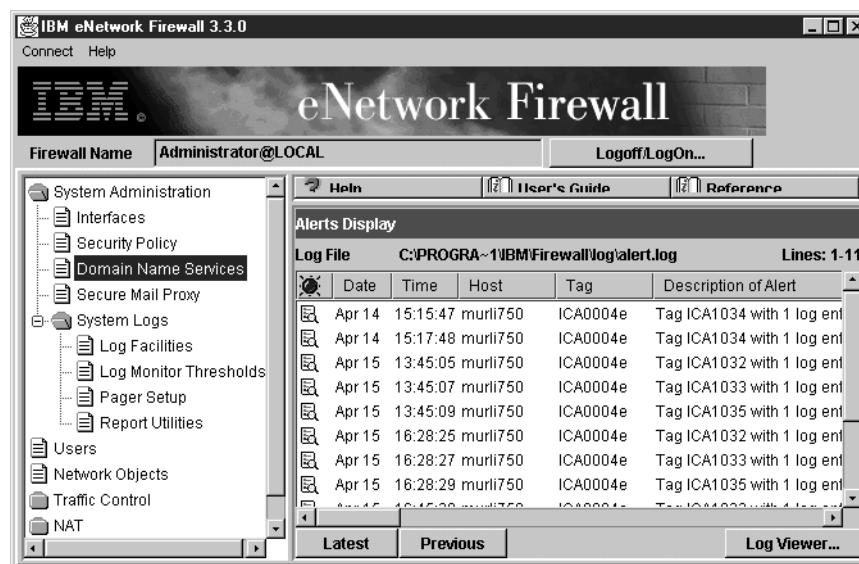


Figure 246. Domain Name Service in Configuration Client

Double-click on **Domain Name Service** to configure the Domain Name Service in the firewall. There will be a warning box that says the DNS configuration file is empty. Click **OK** to close the message box.

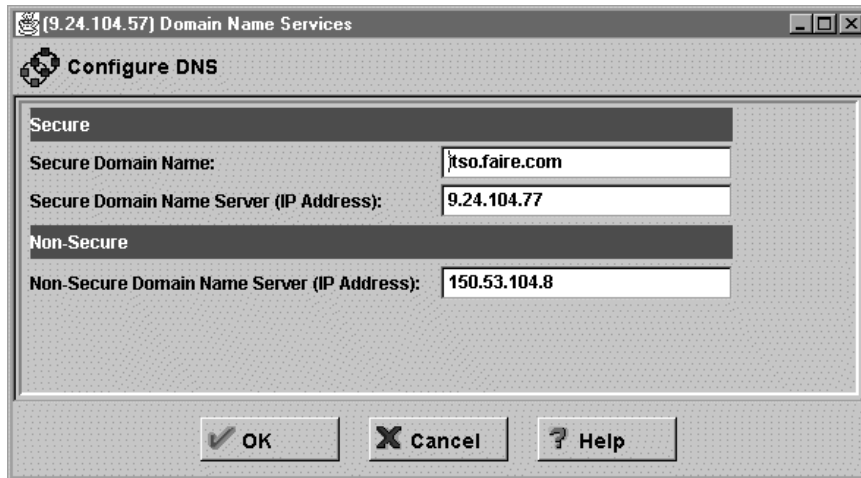


Figure 247. Configure Domain Name Service

Enter your secure domain name and the IP addresses of both internal and external name server machines. We can enter multiple DNS servers in the DNS configuration panel separated by spaces. Click on **OK** and it will finish the configuration of DNS in the firewall.

IBM eNetwork Firewall for Windows NT provides a command line function to setup the DNS configuration file of the firewall. The command name is *fwdns* and its syntax is shown below.

```
fwdns cmd=list

fwdns cmd=change  secdomain="Secure Domain Name"
                  <secaddr="IP address of Secure Name Server">
                  <remaddr="IP address of Nonsecure Name Server">
```

Figure 248. The Syntax of *fwdns* Command

Attention

If we double-click on **Domain Name Service** and the firewall cannot find either the internal or the external server, then the configuration client will hang. Please make sure that both DNS servers are running and if you change the IP address of the internal/external DNS, make sure that the default DNS of the firewall is pointed to the right IP address.

Also make sure that you have an A record for the firewall's secured host name or at least CNAME record, so the firewall can resolve its own interface. Otherwise the configuration client will hang when you try to open **Domain Name Service** configuration in the firewall.

As discussed before, the firewall is a caching-only name server, so there is not much information in the configuration files.

```

;
; Cache file:
;
.                2163095040      IN      NS      externaldns.150.53.104.8
externaldns.150.53.104.8. 2163095040      IN      A       150.53.104.8

```

Figure 249. c:\winnt\system32\dns\fwnamed.ca File on the Firewall

The file above consists of the external name server information. Based on this information, the firewall knows how to get information on external hostnames to response DNS requests submitted by internal hosts.

```

; Created by IBM Firewall 1998030154
@ IN SOA ntf.itso.faire.com. Administrator.ntf.itso.faire.com. (1998030154 3600
600 3600000 86400 )
IN NS ntf.itso.faire.com.
1 IN PTR localhost.

```

Figure 250. c:\winnt\system32\dns\fwnamed.loc File on the Firewall

Since the firewall is a caching-only name server, the file above contains no records.

```

;
; Cache file:
;
.                2163095040      IN      NS      A.ROOT-SERVERS.NET.
A.ROOT-SERVERS.NET. 2163095040      IN      A       198.41.0.4
.                2163095040      IN      NS      B.ROOT-SERVERS.NET.
B.ROOT-SERVERS.NET. 2163095040      IN      A       128.9.0.107
.                2163095040      IN      NS      C.ROOT-SERVERS.NET.
C.ROOT-SERVERS.NET. 2163095040      IN      A       192.33.4.12
.                2163095040      IN      NS      D.ROOT-SERVERS.NET.
D.ROOT-SERVERS.NET. 2163095040      IN      A       128.8.10.90
.                2163095040      IN      NS      E.ROOT-SERVERS.NET.
E.ROOT-SERVERS.NET. 2163095040      IN      A       192.203.230.10
.                2163095040      IN      NS      F.ROOT-SERVERS.NET.
F.ROOT-SERVERS.NET. 2163095040      IN      A       39.13.229.241
.                2163095040      IN      NS      G.ROOT-SERVERS.NET.
G.ROOT-SERVERS.NET. 2163095040      IN      A       192.112.36.4
.                2163095040      IN      NS      H.ROOT-SERVERS.NET.
H.ROOT-SERVERS.NET. 2163095040      IN      A       128.63.2.53
.                2163095040      IN      NS      I.ROOT-SERVERS.NET.
I.ROOT-SERVERS.NET. 2163095040      IN      A       192.36.148.17

```

Figure 251. c:\winnt\system32\dns\cache.dns File on the Firewall

The file above consists of the list of root name servers, which is the same as the cache file of the external name server.

```

; Created by IBM Firewall 1998029110
cache      .                fwnamed.ca
primary    0.0.127.in-addr.arpa fwnamed.loc

```

Figure 252. c:\winnt\system32\dns\boot file on the Firewall

The file above is the DNS boot file of the firewall. It consists of information about the cache file and the primary name server.

11.4.4 Testing The DNS Configuration

We tested the name resolution from the secure and the non-secure side. From the secure side, as you can see, we could resolve the non-secure side machines both in the DMZ and Internet (securez.com). All commands in Figure 253 are executed in the firewall. The same results will be produced if we execute the same commands from one of the internal machines.

```
C:\>nslookup
Default Server: dns.itso.faire.com
Address: 9.24.104.77

> r2210.faire.com
Server: dns.itso.faire.com
Address: 9.24.104.77

Non-authoritative answer:
Name      r2210.faire.com
Address: 150.53.104.57

> www.securez.com
Server: dns.itso.faire.com
Address: 9.24.104.77

Non-authoritative answer:
Name      www.securez.com
Address: 192.168.51.4
```

Figure 253. Testing DNS Configuration from an Internal Machine

From the external network we can resolve only machines that are located on DMZ or the Internet (securez.com). We cannot retrieve information of the internal hosts since it is already blocked by the firewall. The following commands are executed in dns.faire.com (150.53.104.8)

```
C:\>nslookup
Default Server: dns.faire.com
Address: 150.53.104.8

> r2210.faire.com
Server: dns.faire.com
Address: 150.53.104.8

Name      r2210.faire.com
Address: 150.53.104.57

> www.securez.com
Server: dns.faire.com
Address: 150.53.104.8

Name      www.securez.com
Address: 192.168.51.4

> dns.itso.faire.com

DNS request timed out.
  timeout was 2 seconds.
DNS request timed out.
  timeout was 4 seconds.
DNS request timed out.
  timeout was 8 seconds.
*** dns.faire.com can't find dns.itso.faire.com: Non-existent domain
```

Figure 254. Testing DNS Configuration from External Network

Chapter 12. Mail Handling

One of the reasons why an organization would want to connect its network to the Internet is mail access. Through the Internet, people in an organization can exchange mail with the rest of the world. IP mail is transmitted via the SMTP protocol, which is a simple client/server architecture allowing store and forward or direct delivery. In recent years, Lotus Notes has become very popular as a mail solution as well as a workgroup solution.

Usually, people want to have free access in and out of the secure network for mail traffic. The idea of a mail relay is to avoid a direct session from being carried across the firewall gateway. It also hides the internal mail gateway from the nonsecure network. Only the firewall mail server is advertised outside the secure network, which is much more resistant to attack than the real mail gateway.

IBM eNetwork Firewall for Windows NT uses its own secure mail gateway called *Secure Mail Proxy*. Secure Mail Proxy obstructs attempts to subvert the mail server by validating that certain SMTP commands are properly formed before they are relayed to the secure mail server. A user who telnets to IBM eNetwork Firewall for Windows NT on port 25 is unable to damage the server.

12.1 Functionality of Secure Mail Proxy

Secure Mail Proxy is implemented by an NT service named *IBM Firewall Secure Mail Proxy Server*. It is automatically started when we start the firewall. Secure Mail Proxy acts like a mail exchanger, but it does not queue any inbound or outbound mail nor does it store mail before formatting it. If a destination server accepts the mail, the mail packets are transferred to the destination server like a bidirectional *pipe*. If a destination server is not available or does not accept the mail, the mail is rejected.

When an SMTP server opens an SMTP conversation with the firewall's Secure Mail Proxy, the SMTP conversation takes place between the proxy and the sending server until the sending server sends the list of recipients. Then, as each recipient is sent, the Secure Mail Proxy opens a new SMTP conversation with each of the necessary recipient servers. Then, as the body of the message is sent, it will be fanned out (as it comes in) to each of the recipient servers.

12.1.1 SMTP Commands

Secure Mail Proxy can understand all SMTP commands: EHLO, HELO, MAIL, RCPT, DATA, RSET, SEND, VRFY, EXPN, HELP, NOOP and QUIT. Before receiving an RCPT command, Secure Mail Proxy accepts EHLO, HELO, MAIL, RSET, HELP, NOOP and QUIT, but all others are rejected. After it receives the RCPT command it looks for the destination server according to the path defined by RCPT. If found, Secure Mail Proxy tries to connect to the destination mail server at TCP port 25. If the destination mail server cannot be found or a connection cannot be made, Secure Mail Proxy rejects the RCPT command and waits for another RCPT command.

If a connection is made, Secure Mail Proxy transfers subsequent commands (except VRFY and EXPN to a secure mail server) to the destination mail server. VRFY is used to verify a user name in a mail server. The input string is a user

name and the result is detailed information about the user and his/her mailbox. EXPN is used to expand a mailing list. The input string is the mailing list name and the multiple responses will be given containing the full name of the users and their mailboxes. It is dangerous to allow VRFY and EXPN commands because it discloses information of valid e-mail addresses to the outside world.

12.1.2 Multiple Secure Mail Servers

You can set up multiple secure mail servers. Secure Mail Proxy determines the destination secure mail server by looking up the destination specified by the RCPT command. This destination is compared to the domains specified in the Public Domain Name field of the secure mail server setup. If a name is found, Secure Mail Proxy tries to connect to the corresponding mail server.

If a name is not found, Secure Mail Proxy attempts to resolve the name as a hostname. If a hostname is not found, Secure Mail Proxy makes a call to DNS for a mail exchanger (MX record) that corresponds to the name and if found, connects to it. If multiple MX records are found, the Secure Mail Proxy will follow the priority set for those records in DNS.

If an MX record is not found, then the Secure Mail Proxy will try to find an A record.

The Secure Mail Proxy fans out messages to multiple domains, with a limit of 32 domains. The default is 10 and is set by SMTPSB.MAX_FAN_OUT in fwsecuremail.cfg.

12.1.3 Host Name and Domain Name Rewriting

Secure Mail Proxy will replace the secure domain name with the public domain name on all outgoing mail. If they are the same, no domain name hiding will be done! The following headers will be rewritten by Secure Mail Proxy:

- Received: from
- Message-Id:
- From:
- To:
- Reply-To:
- Cc
- Bcc

The next screen shows what Secure Mail Proxy has made, when you choose from the Netscape Mail Messenger screen **View, Headers, all**.

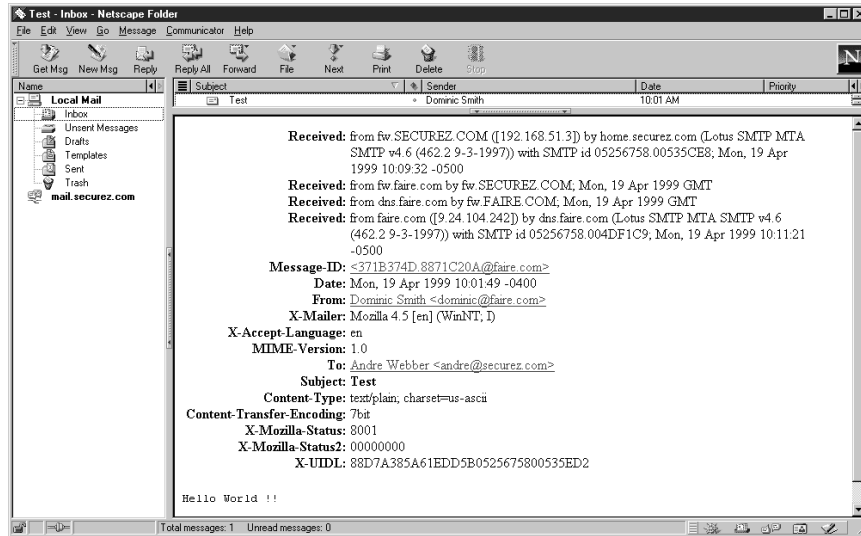


Figure 255. Mail Header View

Secure Mail Proxy does not rewrite inbound addresses set in RCPT commands. So, the internal mail gateway server is responsible for relaying inbound mail to the appropriate destination. The internal mail gateway must be configured so as to accept the public domain name as an alias for their private domain names.

12.2 How Does Secure Mail Proxy Work?

The Secure Mail Proxy acts as a real-time gateway between two or more e-mail domains. In contrast with a traditional SMTP relay, messages are not stored on the firewall before being forwarded to the destinations. The SMTP conversation is interpreted as it happens, and the Secure Mail Proxy conversation is forwarded on to each of the necessary destination servers, command by command. For both incoming and outgoing mail, it only relays the message based on the Secure Mail Proxy configuration file: `c:\Program Files\IBM\Firewall\config\fwsecuremail.cfg`.

The configuration of Secure Mail Proxy is discussed in 12.3, “Configuring Mail Handling” on page 278.

12.2.1 The Overflow Server

The overflow server is responsible for handling any messages which, due to errors, the proxy was unable to handle. Messages are routed to the overflow server under these circumstances:

1. One or more receiving SMTP servers generate an error after the proxy begins to transmit the body of the message, while one or more receiving SMTP servers involved in the same transmission receive the message successfully.
2. The note being sent exceeds the proxy's fan-out limit. The proxy will only open a certain number of outbound connections for delivering the message to each destination domain's mail server (fan-out). Destinations exceeding that limit will be forwarded to the overflow server. This fan-out limit has a default value of 10 as seen in the SMTPSB.MAX_FAN_OUT parameter of the `fwsecuremail.cfg` file.
3. If the destination server is known to DNS but it is not available, then the message is sent to the overflow server.

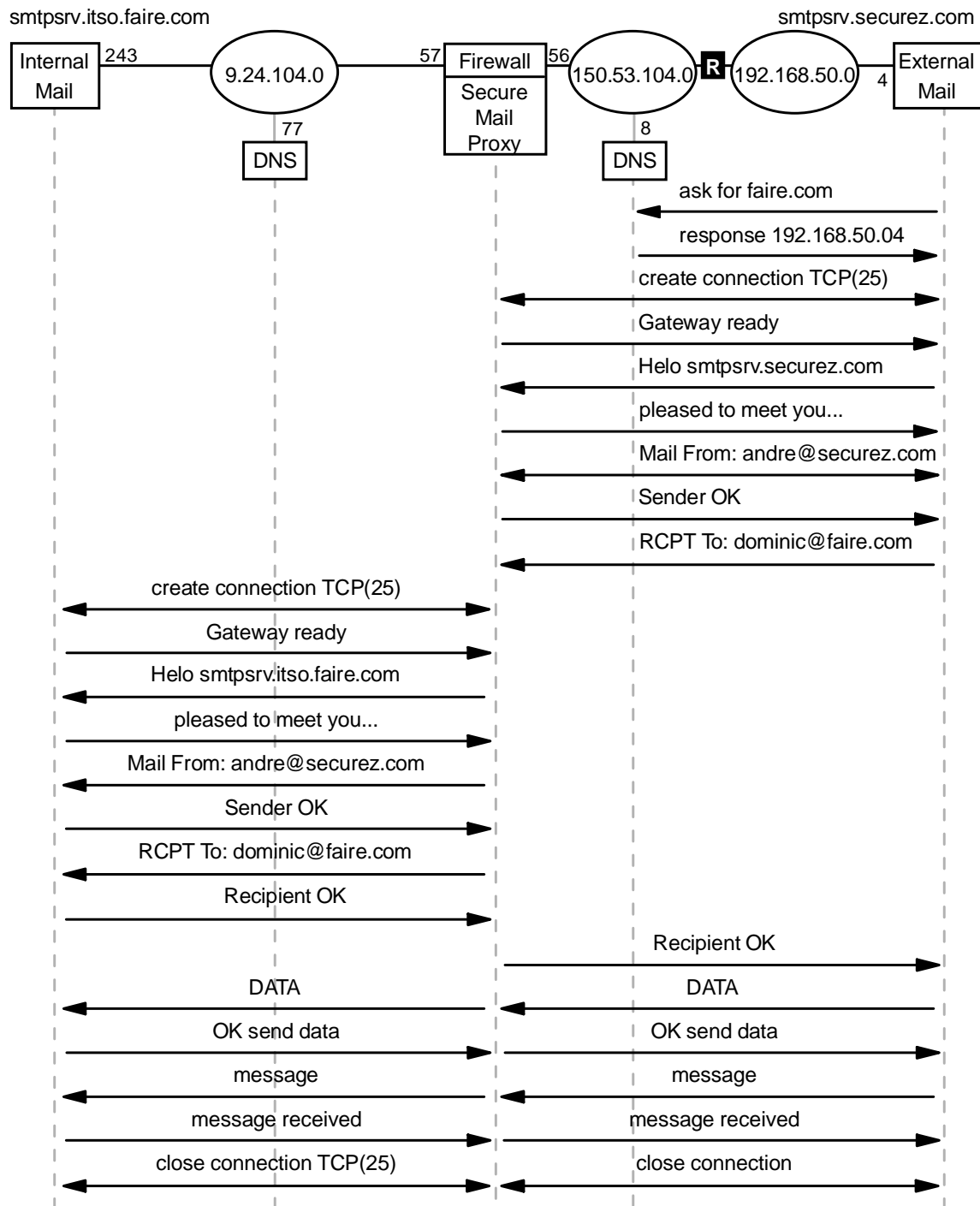
| The overflow server can be co-resident with the firewall, or it can reside on a
| different computer. However, it should be placed on the secure side of the
| network.

| The configuration of the overflow server is discussed in 12.4, "Configuring an
| Overflow Server" on page 283.

12.2.2 Incoming Mail Flow

First, we will look at incoming mail. Figure 256 on page 273 below illustrates the flow of incoming mail.

Incoming Mail



5209\pmxh015

| Figure 256. Incoming Mail Flow

We assume that mail has been sent from the client to the SMTP server, so the flow will begin from the SMTP server. The flow is based on the configuration that is installed in our lab. The mail is sent from andre@securez.com to dominic@faire.com.

Step 1 The external mail server tries to resolve the destination domain. It asks the external DNS for the destination domain. It can't be resolved by the name server, but the name server has an MX record which points to the IP address of the nonsecure interface of the firewall. Figure 257 on page 274 is an example of the MX record.

faire.com	IN	MX	10	nonsecure.faire.com.
nonsecure.faire.com	IN	A	150.53.104.56	

Figure 257. Example of MX Record

- Step 2** The name server sends a response to the external mail server. The response contains the IP address of the non-secure interface of the firewall. The IP address will be used by the external mail server to create an SMTP session.
- Step 3** The external mail server and nonsecure interface of the firewall create a TCP session on port 25 (SMTP). In Figure 256 on page 273, we don't give a detail flow of how the TCP connection is established. As the you know, it is established by a three-way handshake process.
- Step 4** The external mail server sends the SMTP message: HELO smtpsrv.securez.com.
- Step 5** Secure Mail Proxy answers with pleased to meet you....
- Step 6** The external mail server sends MAIL From:andre@securez.com. This describes the source address of the mail.
- Step 7** Secure Mail Proxy accepts the sender by replying Sender OK.
- Step 8** The external mail server sends RCPT To:dominic@faire.com. This describes the destination address of the mail.
- Step 9** Before Secure Mail Proxy replies to the message, it will try to resolve the destination address first. Based on the configuration file of Secure Mail Proxy, it knows where the TCP connection should be established. In this case the internal mail server address is the destination IP address.
- Step 10** Secure Mail Proxy creates a TCP connection on port 25 from a secure interface to the IP address of the internal mail server.
- Step 11** The internal mail server sends the response Gateway Ready, explaining that the SMTP gateway is ready.
- Step 12** Secure Mail Proxy sends the SMTP message: HELO m. smtpsrv.itso.faire.com.
- Step 13** The internal mail server answers with pleased to meet you....
- Step 14** Secure Mail Proxy starts to send the source address of the mail by sending MAIL From:andre@securez.com.
- Step 15** The internal mail server will answer with Sender OK.
- Step 16** Secure Mail Proxy sends RCPT To:dominic@faire.com to describe the destination address of the mail.
- Step 17** The internal mail server checks the domain name that is sent by Secure Mail Proxy. If it matches the domain name list of the internal mail server configuration, then it will reply Recipient OK. Otherwise, it will

reject the SMTP message by sending domain `faire.com` unknown. It will directly impact the response that should be sent by the Secure Mail Proxy to the external mail server.

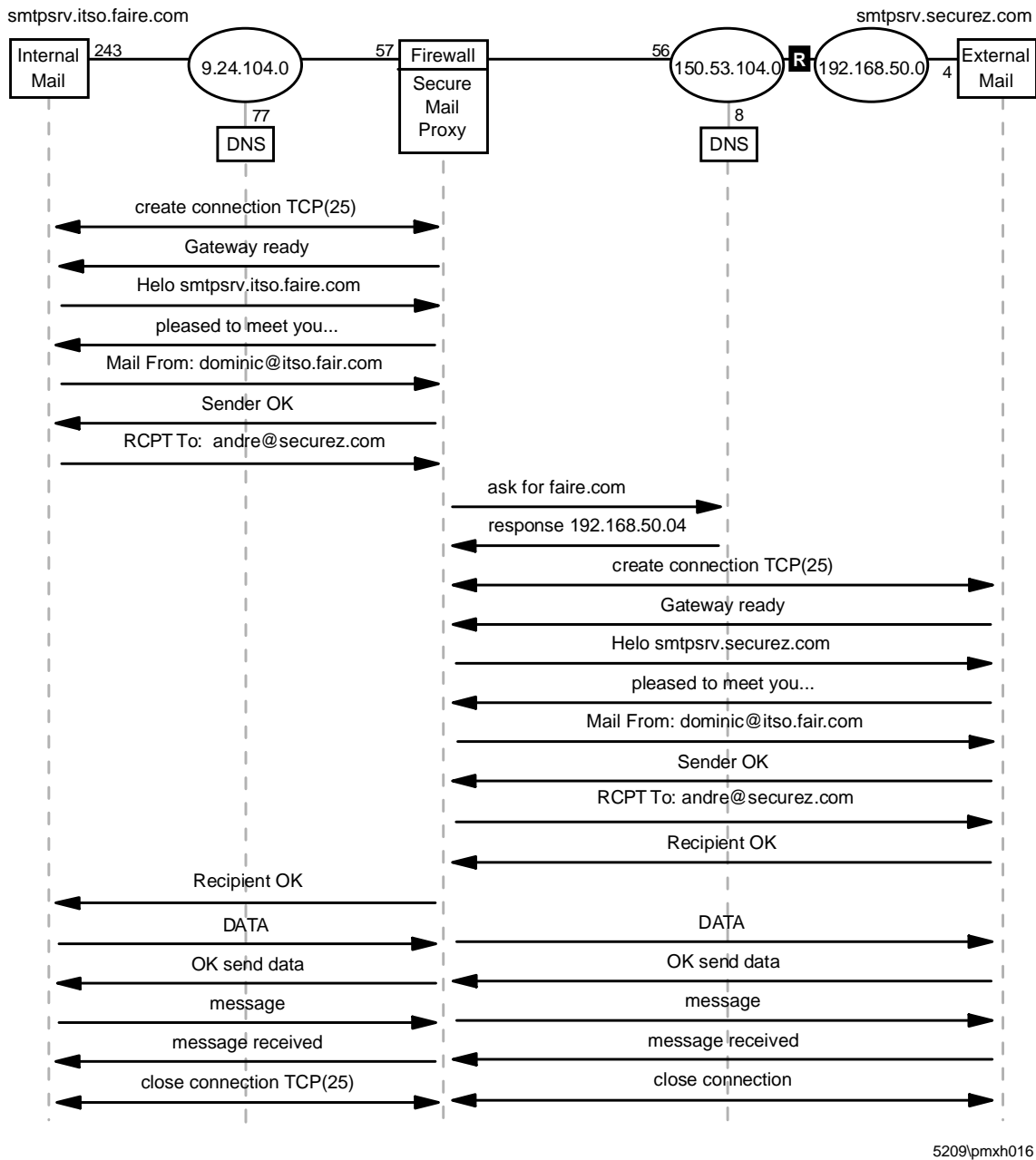
Therefore, it is very critical to set up the internal mail server as a gateway of all domains in secure network.

- Step 18** Secure Mail Proxy will also send a Recipient OK message to the external mail server.
- Step 19** The external mail server sends the SMTP command: DATA to ask whether Secure Mail Proxy is ready to receive data. And SafeMail also sends the SMTP command DATA to ask whether the internal mail server is ready to receive data.
- Step 20** The internal Mail server responds with OK send data to Secure Mail Proxy, which indicates that the internal mail server is now ready to receive data. That response is also sent from Secure Mail Proxy to the external mail server for the same purpose.
- Step 21** Now, the data will flow from the external mail server to the internal mail server through Secure Mail Proxy. This is like an application proxy.
- Step 22** The data will be relayed line by line and every time the internal mail server receives one line, it will acknowledge the data by sending the SMTP message message received. This acknowledgment will be sent also from Secure Mail Proxy to the external mail server.
- Step 23** After all messages have been sent, both the external mail server and Secure Mail Proxy will respectively close the TCP connections with SafeMail and the internal mail server.

12.2.3 Outgoing Mail Flow

Outgoing mail basically uses the same concept as for incoming mail. But now, Secure Mail Proxy should resolve the external mail server. The mail is sent from `dominic@faire.com` to `andre@securez.com`. The outgoing mail flow is illustrated in Figure 258 on page 276.

Outgoing Mail



| Figure 258. Outgoing Mail Flow

- Step 1** First, we have to make sure that the configuration in the internal mail server/gateway is right. The internal mail server/gateway should relay all mail that is destined to the Internet to Secure Mail Proxy. Then, it will create a TCP connection on port 25 with the secure interface of the firewall.
- Step 2** Secure Mail Proxy will answer the connection by sending the SMTP message Gateway ready.
- Step 3** The internal mail server begins the SMTP conversation by sending HELLO smtpsrv.itso.faire.com.

- Step 4.** Secure Mail Proxy will reply with `pleased to meet you.`
- Step 5** The internal mail server sends the source address of the mail using the SMTP message `MAIL From:dominic@itso.faire.com.`
- Step 6** Secure Mail Proxy answers the message with `Sender OK.`
- Step 7** The internal mail server sends the destination address of the mail using the SMTP message `RCPT To:andre@securez.com.`
- Step 8** Secure Mail Proxy will resolve first the destination domain from the destination address of the mail. This resolution will use the standard flow of the firewall. It will ask the internal name server first; if the internal name server doesn't know the answer, it will forward the request to the firewall. The firewall will ask the external name server; if the external name server cannot find an A record for this name, then there is no valid answer for Secure Mail Proxy. Normally the DNS would answer with the address defined in an MX record.
- We suppose that the external DNS finds an A record for this address. The external name server sends the response to firewall, the firewall sends it to the internal name server and the internal name server sends it back to the firewall. Now, the firewall knows the destination IP address of the external mail server.
- Step 9** Secure Mail Proxy creates a TCP connection on port 25 with the external mail server.
- Step 10** The mail server responds to the connection by sending the SMTP message: `Gateway ready.`
- Step 11** Secure Mail Proxy starts the conversation with the SMTP command: `HELO smtpsrv.secure.com.`
- Step 12** The external mail server sends a response `pleased to meet you... ,` which indicates that the external mail server is ready to begin the conversation.
- Step 13** Secure Mail Proxy sends the source address of the mail by sending the SMTP command `MAIL From:dominic@faire.com.` Please notice that Secure Mail Proxy has changed the source domain from `itso.faire.com` to `faire.com`. This is another major function of Secure Mail Proxy: to hide internal domain names.
- Step 14** The external mail server sends a response `Sender OK` to indicate that it accepts the sender.
- Step 15** Secure Mail Proxy sends `RCPT To:andre@securez.com` to describe the destination address of the mail.
- Step 16** The external mail server will determine whether this is the valid destination domain name or not. If it is the valid domain name, then the external mail server will respond with the `Recipient OK` message.
- Step 17** Secure Mail Proxy will also send the same message (`Recipient OK`) to the internal mail server, which indicates that the destination name is well known.
- Step 18** The internal mail server sends the SMTP command: `DATA` to ask whether Secure Mail Proxy is ready to receive data. And SafeMail also sends the same SMTP command to the external mail server.

- Step 19** The external mail server responds with OK send data to Secure Mail Proxy, which indicates that the external mail server is now ready to receive data. That response is also sent by Secure Mail Proxy to the internal mail server for the same purpose.
- Step 20** Now, the data will flow from the internal mail server to the external mail server through Secure Mail Proxy.
- Step 21** Secure Mail Proxy will relay the data line by line and every time the internal mail server receives the data, it will acknowledge the data by sending the SMTP message message received to Secure Mail Proxy. This acknowledgement is also sent by Secure Mail Proxy to the external mail server.
- Step 22** After all messages have been sent, both the internal mail server and Secure Mail Proxy will respectively close the TCP connections with SafeMail and the external mail server.

12.3 Configuring Mail Handling

The configuration of mail handling in a firewall requires the centralized mail handlers. These centralized mail handlers are referred to as your SMTP gateways or secure mail servers. All incoming/outgoing mail will flow through these SMTP gateways and the firewall. We assume that you already have a centralized mail gateway.

First we have to make sure that all DNS configurations are well set. For outgoing mail, the firewall should be able to resolve external domains since Secure Mail Proxy on the firewall will use it to send mail. Also for outgoing mail, the centralized SMTP gateway (internal SMTP server) should be set to relay all outgoing mails to firewall.

For incoming mail, the external DNS should have at least two MX records which point to the IP address of the nonsecure interface of the firewall and another to a backup mail server. Below is a sample of MX records in the external DNS. See Figure 256 on page 273 and Figure 258 on page 276 for examples of mail configuration.

faire.com	IN	MX	10	nonsecure.faire.com.
faire.com	IN	MX	20	smtpsrv2.someisp.net
nonsecure.faire.com	IN	A	150.53.104.56	

Figure 259. Example of MX Records in the External Mail Server

The above DNS records show that we have two mail servers that will be pointed to by faire.com. The first one is the nonsecure interface of the firewall and the second is a backup mail server.

We have also added forwarders on the internal name server. The forwarders should point to the secure interface of the firewall in order to enable DNS to do name resolution for external hostnames through the firewall. For a detailed discussion on DNS please refer to Chapter 11, "Domain Name Service (DNS)" on page 247.

Now, we are ready to set up our mail relay in the firewall.

Open the configuration client and log on to the firewall. Double-click on **System Administration** to show the menu and double-click on **Security Policy** to open the Security Policy window.

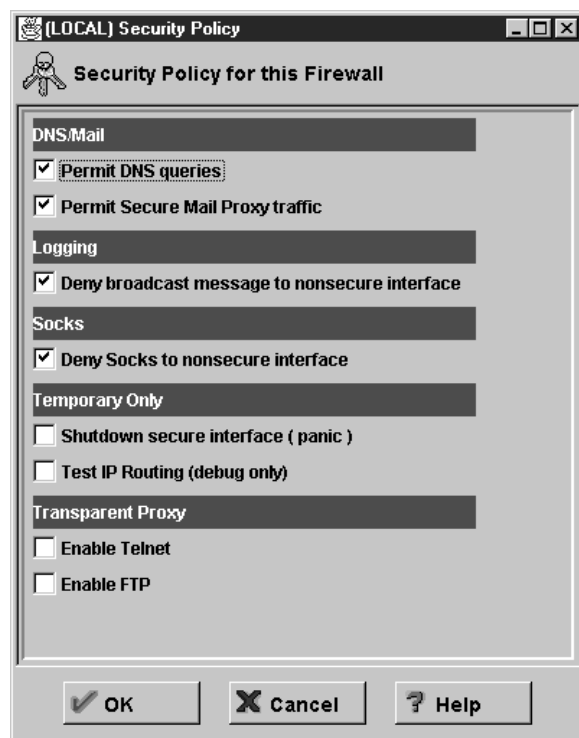


Figure 260. Security Policy

Check **Permit Secure Mail Proxy traffic** to enable the Secure Mail Proxy. Click on **OK** to save the configuration. The message box appears to ask whether we want to refresh the connection rule instantly. When we turn on the Secure Mail Proxy traffic, the firewall should add some connections to enable SMTP relay.



Figure 261. Information Box on Activating Connection

Click on **Yes** to access the activation panel. Click on **Execute** to regenerate connection rules and activate them.

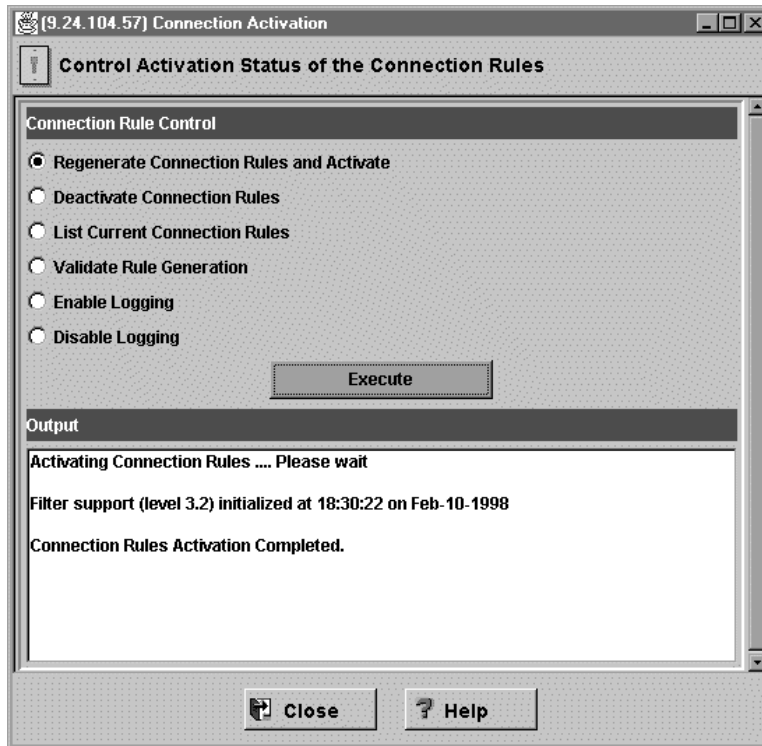


Figure 262. Connection Activation Window with Output

The status of activating the connection rules will be displayed. After completing the connection rules activation, click on **Close** to return to the main menu.

See 11.4.3, "Configuration of DNS in Firewall" on page 263 for details on DNS setup. For a discussion on creating connections, services and rules, please see Chapter 9, "Examples of Rules for Specific Services" on page 161.

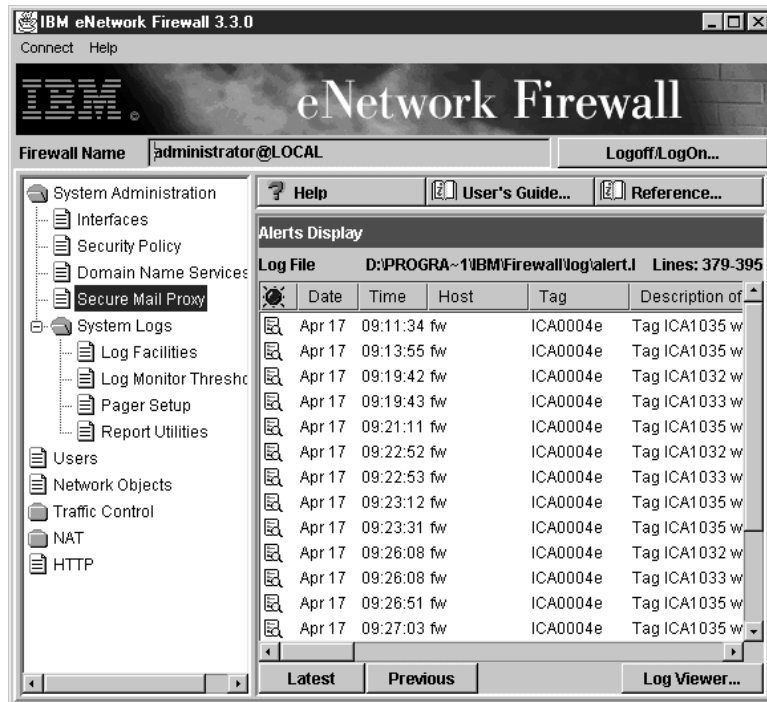


Figure 263. Configuration Client Main Screen

Now we are going to configure Secure Mail Proxy. Double-click on **Secure Mail Proxy** from the System Administration menu to open the Secure Mail Proxy window. Because this is the first time we are configuring Secure Mail Proxy, an error message box appears and tell us that the definition file of Secure Mail Proxy is empty.

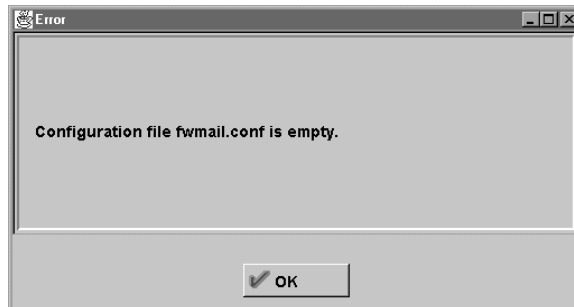


Figure 264. Error Message of Mail Configuration File

Ignore the box by clicking on **OK**.

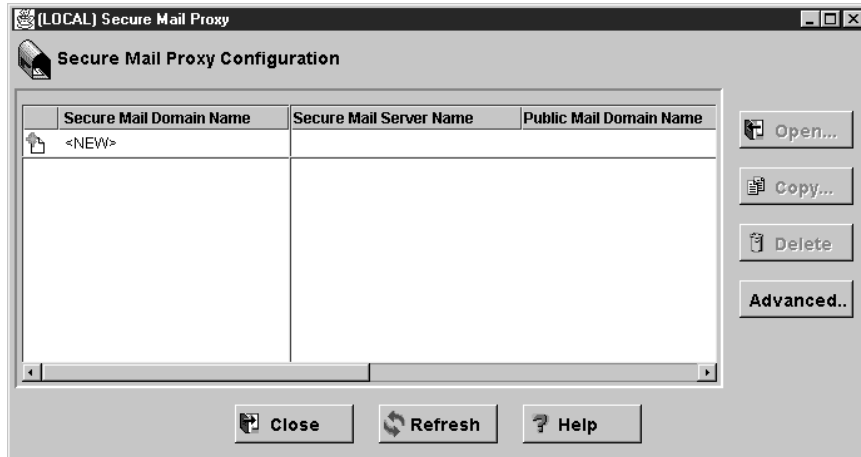


Figure 265. Secure Mail Proxy Window

On the Secure Mail Proxy screen, we can see that there is no definition for Secure Mail Proxy. Double-click on **<NEW>** to add a new mail server.

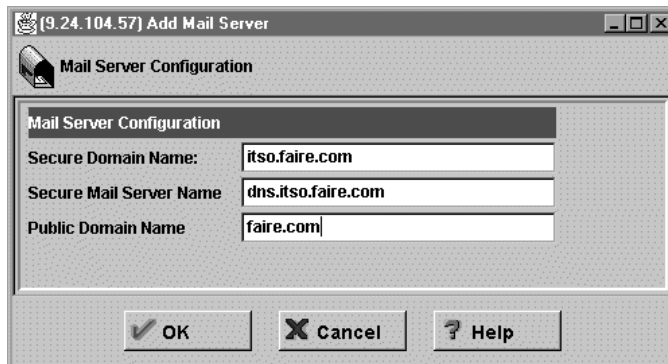


Figure 266. Add Mail Server Window

Enter the Secure Domain Name, which is the domain name of your secure network, the Secure Mail Server name, which is the host name of your internal SMTP gateway and the Public Domain Name, which is the domain name of your nonsecure network. Click on **OK** to save the changes.

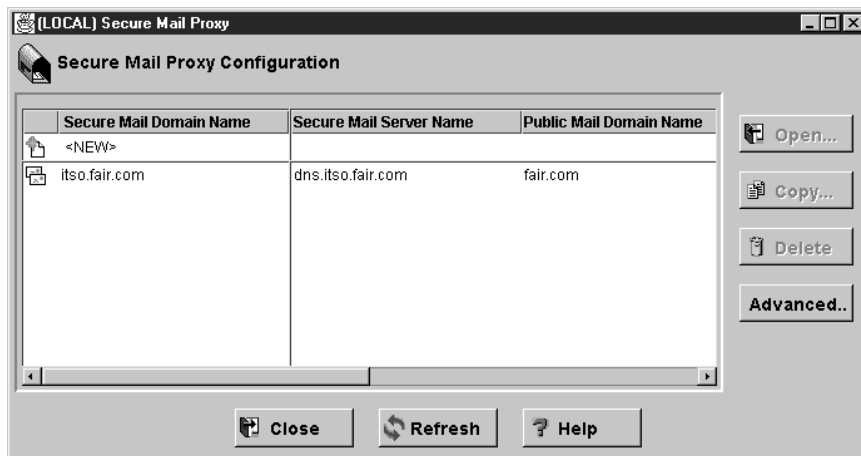


Figure 267. Secure Mail Proxy Window with New Item

As you can see from Figure 267 there is a new item in the Secure Mail Proxy configuration. Using this dialog box, you can add as many Secure Mail Proxy definitions as needed to support your own environment. Click on **Close** to save the changes and return to the main screen. This will finish the configuration of Secure Mail Proxy in the firewall.

Note: If you take a look at Figure 255 on page 271, you will see that the recipient is able to see the secure IP address of the sender (192.168.51.3). If you want to avoid this, we should define all the IP addresses we want to hide as Secure Mail Server Name, and the nonsecure address of the firewall as the public domain name in Figure 266 on page 282. It is not possible to define subnets; it has to be IP addresses.

IBM eNetwork Firewall for Windows NT provides a function to modify the Secure Mail Proxy configuration file using a command line interface. The command name is `fwmail` and the syntax is shown in Figure 268.

```
fwmail cmd=list

fwmail cmd=add      secdomain="Secure Domain Name"
                   mail="Secure Mail Server Name"
                   remdomain="Public Domain Name"

fwmail cmd=change  secdomain="Secure Domain Name"
                   <mail="Secure Mail Server Name">
                   <remdomain="Public Domain Name">

fwmail cmd=delete  secdomain="Secure Domain Name"
```

Figure 268. `fwmail` Command Syntax

12.4 Configuring an Overflow Server

The overflow server should be placed on the secure side of the network.

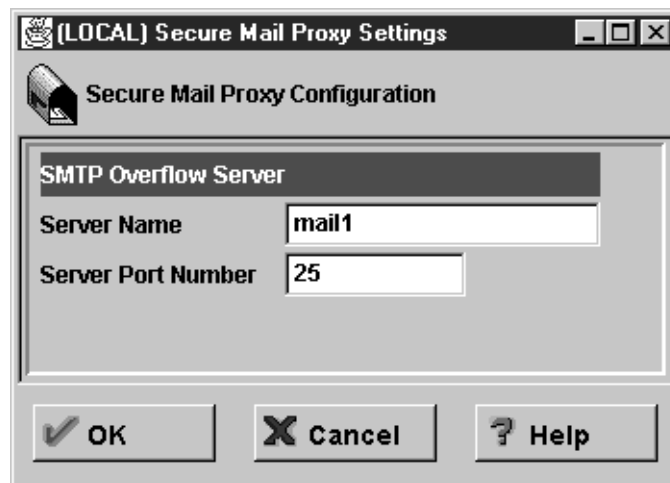


Figure 269. Add Overflow Server Window

To set up the overflow server, click **Advanced** on the Secure Mail Proxy panel as seen in Figure 267 on page 282. You are prompted for the server name (or IP address) and port number for the overflow server. If you provide the IP address of the server instead of its name, enclose the address in square brackets.

You can use "localhost" or "127.0.0.1" if the overflow server resides on the firewall machine. This IP address is not filtered by the firewall's traffic control. Configure the overflow server to listen on a non-standard port (for example, 2500 instead of 25). The overflow server must be configured as a store-and-forward mail relay, having the firewall proxy as its outbound gateway; otherwise, a message could be forced to the overflow server as a means of bypassing the Firewall's proxy.

We installed the overflow server in the firewall machine, but probably you will not want to do that, in order to comply with the principle stated about the firewall machine: KISS (keep it simple...).

Click **OK** to save your changes, or **Cancel** if you change your mind.

12.5 Tailoring the Secure Mail Proxy

To tailor the Secure Mail Proxy, a configuration file exists in c:\Program Files\IBM\Firewall\config\fwsecuremail.cfg. This file controls the behavior of the Secure Mail Proxy with many configuration parameters.

You certainly are not encouraged to change this file as the default settings are in accordance for normal mail relay situations. Several functions described in this file have not been implemented in the GA version of the IBM eNetwork Firewall for Windows NT 3.3; they will be fully implemented in a future version of the firewall. On top of that, if you run into trouble by changing these parameters, the firewall support team will not help you as these are unsupported features.

Next, we will show mail handling examples that have been tested in our lab.

12.6 Mail Handling Examples

It is easier to understand the interaction of the different mail configuration elements by looking at examples. We will consider two cases: one is for incoming mail and the other is for outgoing mail. We assume that the interaction between the mail client and mail server is using the Post Office Protocol (POP) for incoming messages. The POP mechanism allows users to check for their mail while mailboxes are maintained in the mail server. Another advantage is that the mail server doesn't have to access a user's mailbox directory which means that the client shouldn't have to have a mailbox in the workstation.

We will give you the step-by-step description of how the messages flow from the client to server for both incoming and outgoing mail. The configuration we have in our lab is shown in Figure 270 on page 285.

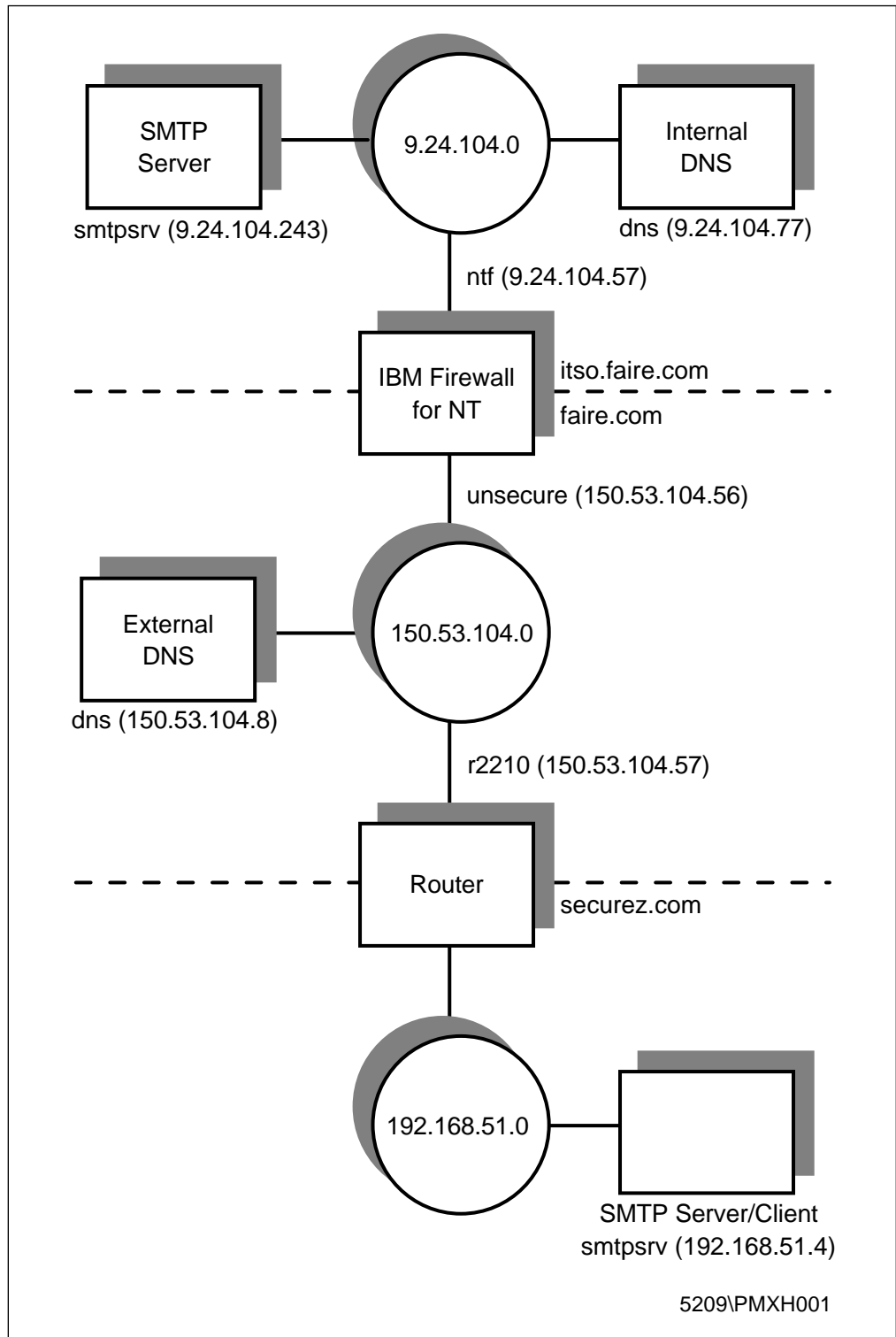


Figure 270. Example of Mail Configuration

We use NT Server 4.0 as the platform for all servers. We use Lotus Notes as an SMTP/POP3 mail server and Netscape Messenger as client running on NT. Now, let's take a look at incoming mail first.

12.6.1 Incoming Mail to the Secure Network

In the first case, we are sending mail from a Netscape client using SMTP accessing to smtpsrv.itso.faire.com as the SMTP server. Actually, we send the mail from the same machine where the external SMTP server resides (that is, smtpsrv.securez.com). Please see Figure 270 on page 285 for our mail configuration example. The mail is sent from smtpsrv.securez.com as an example of an Internet mail server to smtpsrv.itso.faire.com as an internal mail server.

User andre@securez.com wants to send a message to dominic@faire.com. Now we will describe how the mail gets through.

Step 1 Using Netscape mail client, the user on securez.com sends the message to smtpsrv.securez.com as a mail server. It is shown in Figure 271.

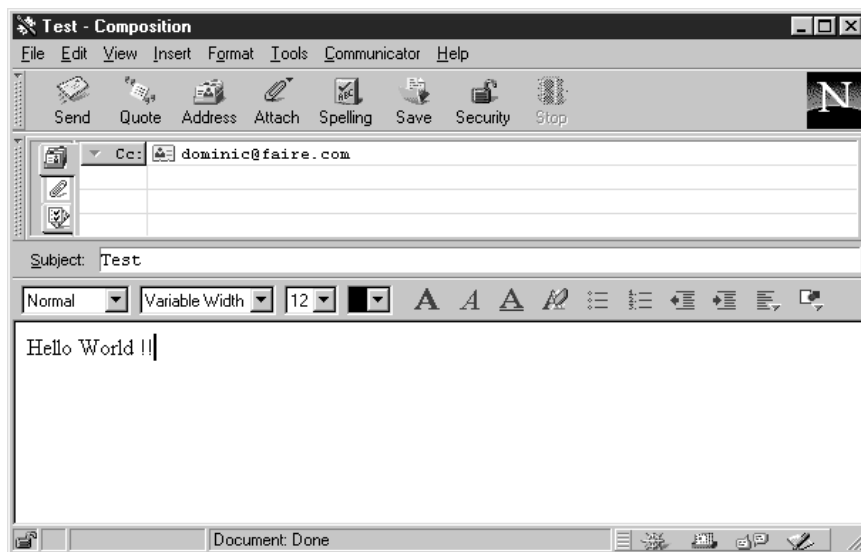


Figure 271. Sending Mail from the Nonsecure Network

Step 2 The SMTP server will try to resolve the destination domain contained in the message header. It uses the following process:

1. The SMTP server on smtpsrv.securez.com asks DNS on dns.faire.com to resolve the IP address of faire.com. In this case, dns.faire.com is the default DNS of securez.com.
2. The request will fail, because dns.faire.com doesn't have an A record for faire.com.
3. Now, the SMTP server will look at the MX record from dns.faire.com. As we mentioned before, there is an MX record in the external DNS, which is shown below:

faire.com	IN	MX	10	nonsecure.faire.com.
-----------	----	----	----	----------------------

Figure 272. MX Record in the External DNS

The MX records specify a mail exchanger for a domain name. In our case, it indicates that mail to faire.com will be forwarded to nonsecure.faire.com.

4. Now, the SMTP server on smtpsrv.securez.com knows how to send messages to the IP address of nonsecure.faire.com.

Step 3 The SMTP server on smtpsrv.securez.com opens a connection to SMTP port (TCP/25) on nonsecure.faire.com instead of faire.com. The mail message is received by Secure Mail Proxy on the firewall.

Step 4 Secure Mail Proxy on the firewall does not store the mail message, but just receives it and reroutes it on a session to the SMTP port on the internal mail server.

The internal mail server smtpsrv.itso.faire.com receives the mail message and saves it to Dominic's mailbox.

Step 5 We will notice new mail in the mailbox when we open the Netscape mail application and connect to smtpsrv.itso.faire.com through POP3. The new message is shown in Figure 273.

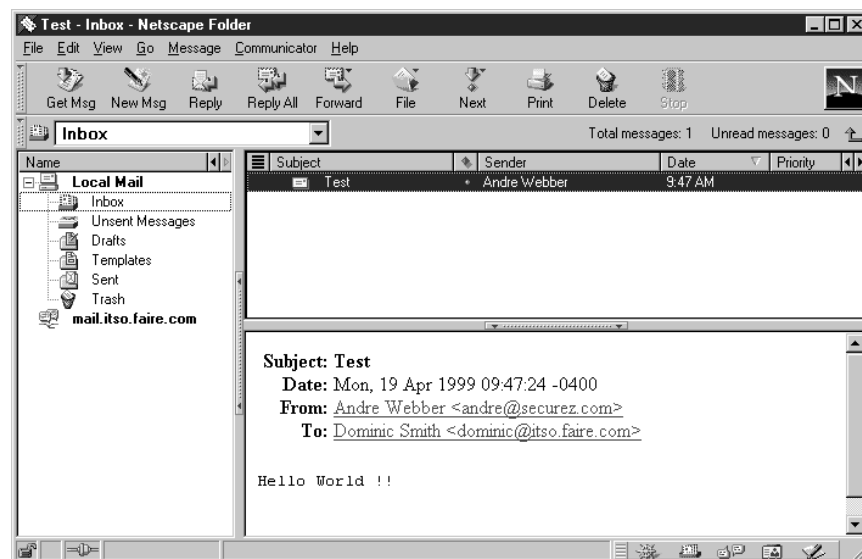


Figure 273. Incoming Mail to the Internal Mail Server

12.6.2 Outgoing Mail to Nonsecure Network

For outgoing mail, we send mail from dominic@smtpsrv.itso.faire.com to andre@securez.com. Please see Figure 270 on page 285 for the mail configuration example. From the SMTP server point of view, we don't have to deal with name resolution since we already set the SMTP server to relay all outgoing mail to the firewall. The firewall will resolve the domain name of the destination address.

Step 1 Using Netscape Messenger, we create mail and through SMTP we send it to the SMTP server on smtpsrv.itso.faire.com. That is shown in Figure 274 on page 288.

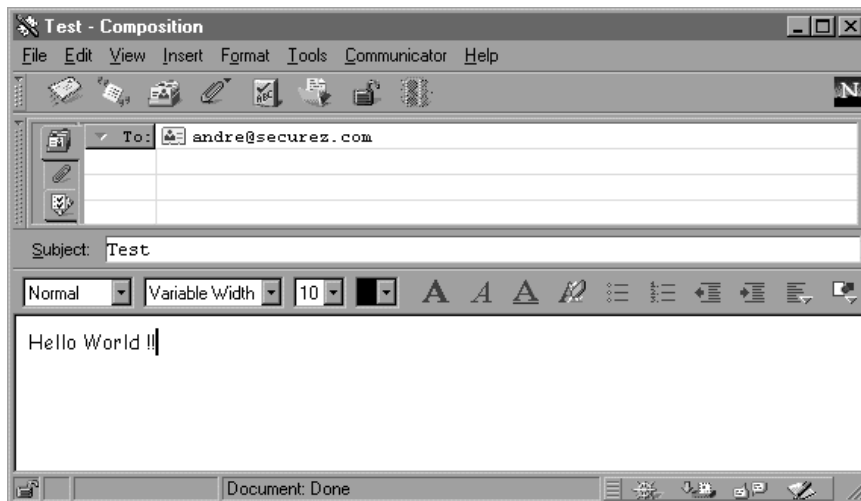


Figure 274. Sending Mail from the Secure Network

- Step 2** Now, the SMTP server on smtpsrv.itso.faire.com opens a connection to SMTP port (TCP/25) on ntf.itso.faire.com (using mail relay). The mail message is received by Secure Mail Proxy on the firewall.
- Step 3** Next, Secure Mail Proxy on the firewall (ntf.itso.faire.com) receives the mail message. Secure Mail Proxy examines the mail headers and removes all *Received* lines from the headers. Secure Mail Proxy converts the internal domain name inside *From* headers to the public domain name. Secure Mail Proxy does not store the mail message, but instead reroutes it to the destination address.
- Step 4** When the firewall receives the SMTP command RCPT To, then the firewall tries to resolve the domain name and follows the standard name resolution procedure. The firewall asks the internal DNS for the destination address. The internal DNS doesn't know and forwards the request to the firewall. The firewall will ask the external DNS and get a response. The response goes back to the internal DNS and the internal DNS sends it to the firewall. For detailed information on DNS please refer to Chapter 11, "Domain Name Service (DNS)" on page 247.
- Step 5** The firewall opens an SMTP connection to TCP port 25 on the IP address of smtpsrv.securez.com. The mail message is received by the SMTP server, which saves it in Andre's mailbox.
- Step 6** We will be notified that new mail has arrived in the mailbox when we open Netscape mail client and connected to smtpsrv.securez.com through POP3 protocol. The new message is shown in Figure 275 on page 289.

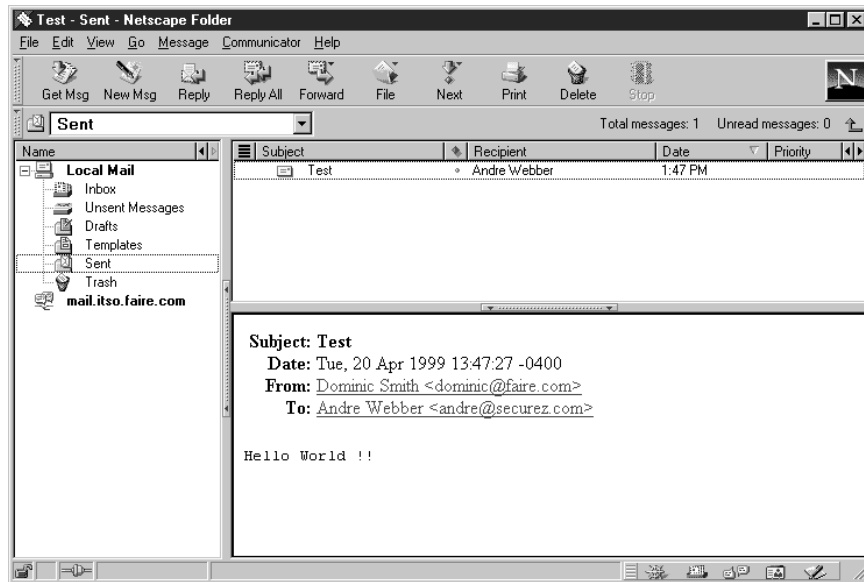


Figure 275. Outgoing Mail to the External Mail Server

12.7 Mail Log

Mail information is logged in the main firewall log. This is the c:\Program Files\IBM\Firewall\log\local4.log file which is shown in Figure 276

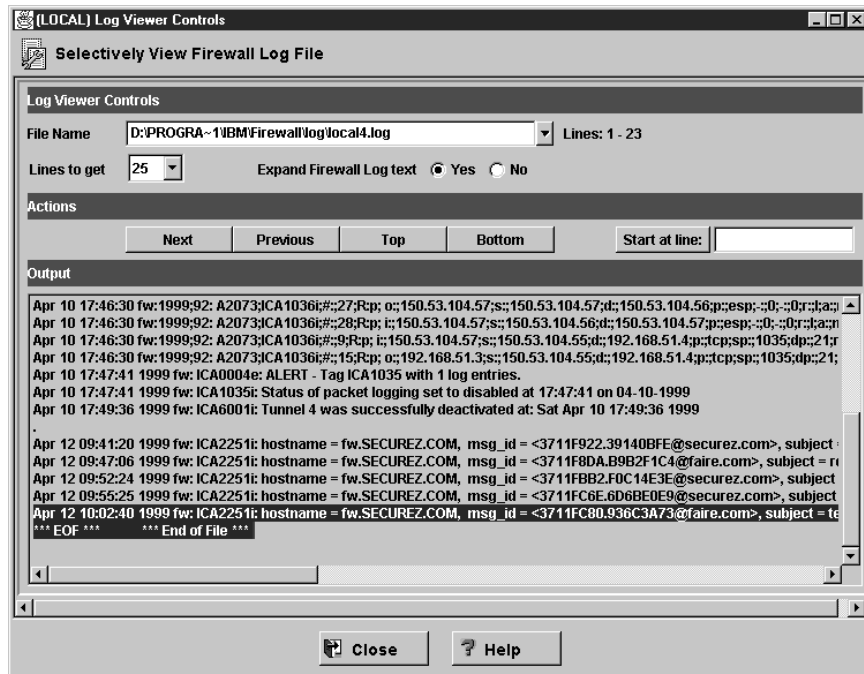


Figure 276. Main Firewall Log

Mail logging which was used to give debug information is not used anymore. The debug logging is done by secure mail specific log facility which is not publicized yet. These log files can be seen in the c:\Program Files\IBM\Firewall\log directory. They are named log(x).dat and log(x).seq where x represents a sequence number from

one to 29. There is no need to erase these files; the firewall will quickly create them again.

These files and the contents of them are not documented at this stage. Despite this fact, during our lab mail testing we found hints regarding mail errors in these files. Feel free to look into these files and investigate, but do not report questions/errors to the firewall support group as this is an unsupported feature. Nevertheless, we will give a few details on these files.

The .seq files contain no useful information for the firewall user; it is for development use only. The .dat files contain the Secure Mail Proxy messages. The format of the messages is as follows:

```
|YYYYMMDD HHMMSS | PID |          Secure Mail Proxy Function |Message
```

Where:

- S is the severity of the message:
 - G General: A basic status message, such as a report on configuration settings.
 - W Warning: A minor error has occurred, but Secure Mail Proxy has automatically recovered.
 - E Error: A serious error has occurred, which has resulted in a failed connection or e-mail.
 - S Security: Secure Mail Proxy has rejected a mail item or connection because of a security check.
 - I Informational: Detailed information about the operation of Secure Mail Proxy, such as a report about the success of an outbound connection.
 - D Debug: Very detailed information about the operation of Secure Mail Proxy, useful primarily to support personnel.
 - T Trace: Messages reporting on the actual step by step execution of Secure Mail Proxy. Useful only to support personnel.
 - i i/o: This level records all SMTP read and write operations. Useful primarily to support personnel.

You can control what kind of severity level messages will be logged by using the variable SMTPSB.LOG_LEVEL in the fwsecuremail.cfg file. Possible choices are:

- (G)eneral
- (I)nformational
- (W)arning
- (E)rror
- (D)ebug
- (T)race
- (i)/o
- (A)ccounting
- (t)iming,
- (S)ecurity

- YYYYMMDD HHMMSS is the date and time the message was logged. Before the date and time there is a prefix internally defined.
- PID is the NT thread ID of the Secure Mail Proxy component emitting the message.
- Message is the text of the diagnostic message.

Some sample messages are:

<i>Table 20. Secure Mail Proxy Sample Error Messages</i>	
Severity	Message
General	G 19971017 093733 32710 Monitor Monitor is Starting.
Warning	W 19971017 093815 35916 ADDR_parse_address()) Could not parse: jayne_doe@east.ispabc.com
Error	E 19971017 093815 35916 SMTP State Machine Connection failed: ISPMail.ispabc.com

Warning!

Do not log off the administrator from the Windows NT operating system; if you do, the Secure Mail Proxy comes to an abnormal end. This problem is addressed already and a fix will be available with refresh 3.3.1 of the IBM eNetwork Firewall for Windows NT.. To overcome this problem, go into Windows NT Services and start the Secure Mail Proxy again. Instead of logging off, use *Lock Workstation* of the Windows NT Security panel.

Chapter 13. Configuring Proxy Services and SOCKS

As we have discussed in 2.2, "IP Filters" on page 12, address-based filters have only a limited use in keeping unwanted intruders away. For a more secure system, we want to break the sessions at the firewall boundary. Application proxies and SOCKS server give us techniques for doing so. And you get another advantage using this techniques. You need no additional registered IP addresses for your internal clients because only the servers on the firewall connecting to the target hosts (not your internal clients) are visible from the Internet.

We still need to use firewall IP filter rules to enforce acceptable modes of access to the firewall. However, these filter rules can be much more restrictive than rules designed to allow traffic directly through the firewall.

There is one major difference between using a proxy or a SOCKS server. For SOCKS you need SOCKSified client versions of your programs or you have to SOCKSify the whole IP Stack. Using a proxy server does not need any modifications on the client program.

All examples discussed in this chapter are based on the scenario in Figure 277. You will also find all IP addresses used in the filter rules examples in this figure.

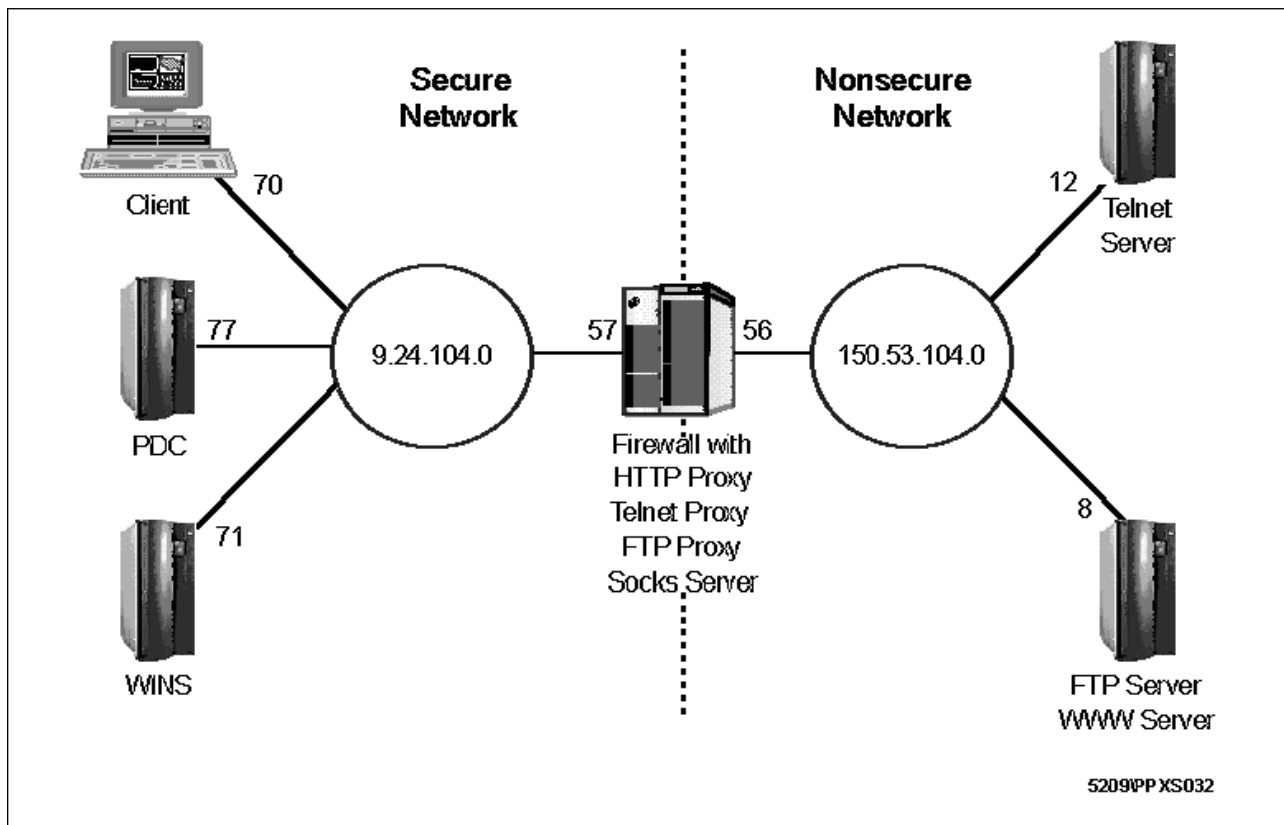


Figure 277. Example Scenario

The figure represents the lab used for our tests. The non-secure network could also be the Internet. The filter rules in the examples already provide access to the Internet.

Some of the text in this chapter that explain the entry fields and valid options on the GUI windows are taken from the *IBM eNetwork Firewall for Windows NT Reference Version 3 Release 3* manual.

13.1 User Administration

This section describes the different user roles and how to add new proxy users to the IBM Firewall.

13.1.1 Types of Users

There are three categories of users defined on an IBM Firewall machine. They are explained in Table 21.

Type	Description
Proxy User	Proxy users are people who use firewall services, such as the HTTP proxy service to access Web sites on the Internet from within a corporate network or the Telnet proxy to establish inbound connections. Proxy users are able to use services through the firewall but do not have access to the firewall machine and cannot perform local logins to the firewall machine.
Firewall Administrators	Firewall administrators can use the firewall proxy services, but they can also configure the firewall by using the configuration client by logging on to the firewall from a remote host. Firewall administrators can create and modify definitions for proxy users but they cannot create or modify the definitions of other firewall administrators.
Primary Firewall Administrators	Have the same capabilities as firewall administrators. They can also perform local logins to the firewall machine. Primary firewall administrators can create and modify definitions for other firewall administrators.

You can create proxy users and firewall administrators with the GUI. You can create primary firewall administrators by creating a local NT account on the firewall and make this account a member of the *administrators group*. Primary and ordinary firewall administrators should not be used to access proxy and SOCKS services.

This chapter covers only proxy users. The configuration of firewall administrators and primary firewall administrators is covered in 10.3, "Creating Administrators" on page 231.

13.1.2 Types of User Databases

There are at least two types of user databases relating with the IBM Firewall: the *firewall user database* and the *Windows NT user database*. They are explained in Table 22 on page 295.

Type	Description
Firewall User Database	This database contains firewall related attributes for all proxy users and administrators. This includes the allowed services and the authentication schemes used.
Windows NT User Database	This database contains any information which is related to NT logon accounts in a Windows NT domain.

Each user who will use services on the firewall needs to have an account either on the IBM Firewall system or on the local NT domain. The information about proxy user is stored in the firewall user database. However, if the proxy user is also defined on the Windows NT user database, the NT logon password of the user can be used if the user requests any service that need to be authenticated.

If the user is not defined in the firewall user database and the user tries to use a firewall proxy service, the default user record `fwdfuser` will be used to define the attributes and authentication schemes used to validate this user (see 13.1.5, "Configuring the Default User" on page 301).

Therefore, it is not necessary to define every proxy user in the firewall user database. You can use the default user record, `fwdfuser` (see Figure 278 on page 296), to assign authentication methods and other attributes to firewall proxy users that are not defined in the firewall user database.

13.1.3 User Authentication Schemes

Table 23 shows all valid choices for user authentication schemes from the authentication pop-up menu in Figure 279 on page 297.

Scheme	Description
Deny all	All user access is denied.
Permit all	No authentication is needed. We recommend that you never use this authentication method from the non-secure side of the firewall.
Firewall Password	The user must be prompted for, and enter, a valid password. This password is stored in the IBM Firewall user database. The firewall password allows more secure passwords and password rules than the NT logon password so this is our recommended choice.
NT Logon	If you choose this method of authentication, your user ID and password will be validated against the local NT user database. NT logon passwords is less secure than the firewall password. However, if users are already defined in an NT domain, you can use the NT logon passwords so the user does not need multiple passwords (see 13.1.6, "Configuring for NT Logon Password Authentication" on page 302).
SecurID Card	Authentication is done using a Security Dynamics SecurID security card or pinpad card. The PIN must be set before using this authentication method with the IBM Firewall (see 21.5, "SecurID Authentication" on page 563).

Table 23 (Page 2 of 2). User Authentication Schemes	
Scheme	Description
User-Supplied	Authentication is supplied by the user. You can install up to three user-supplied authentication methods on the firewall. Please refer to the <i>IBM eNetwork Firewall for Windows NT Reference Version 3 Release 3</i> for more information on how to implement user-supplied authentication methods.

13.1.4 Configuring Proxy Users

The Configuration Client provides several panels to configure proxy users. Adding users is done by opening the **Users** dialog from the navigation tree in the Configuration Client. Figure 278 shows the general user administration window. In this figure you can also see the default user record fwdfuser.

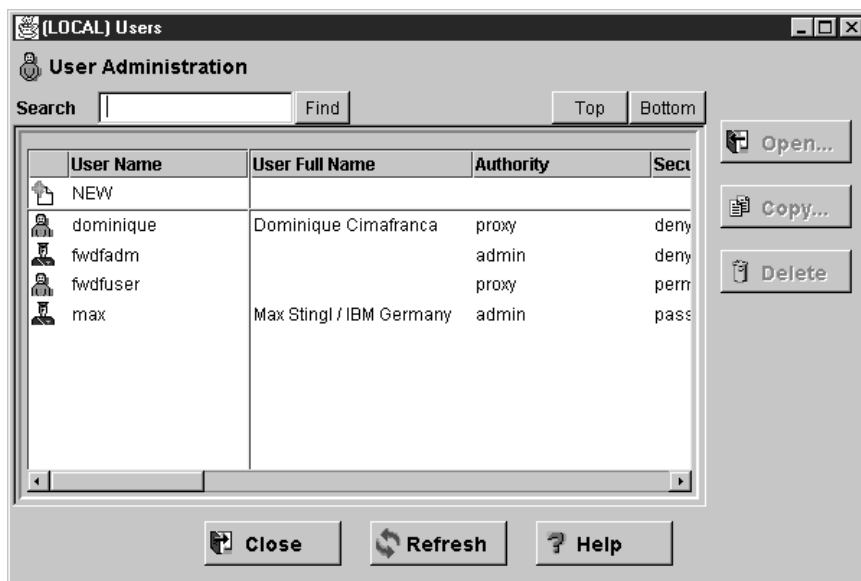


Figure 278. User Administration Window

From this window you can create, modify or delete users. Figure 279 on page 297 shows the dialog for adding a user.

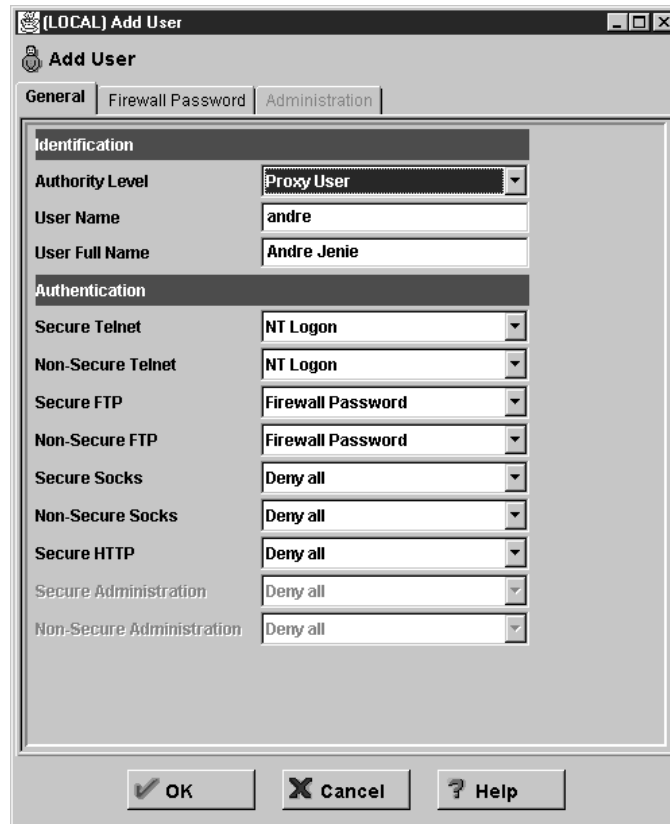


Figure 279. Add User Window

The options you can specify for each user are as follows (the valid user authentication methods for the firewall services are described in 13.1.3, “User Authentication Schemes” on page 295):

Authority Level is either Firewall Administrator or Proxy User (see 13.1.1, “Types of Users” on page 294). In this chapter we explain only the Proxy User option.

User Name is the ID that the user will use to log in to the IBM Firewall.

User Full Name is the textual name of the user. This is used for documentation purposes only.

Secure Telnet indicates the authentication method for a user logging into the Telnet proxy from the secure side of the firewall.

Non-Secure Telnet indicates the authentication method for a user logging into the Telnet proxy from the non-secure side of the firewall.

Secure FTP indicates the authentication method for a user logging into the FTP proxy from the secure side of the firewall.

Non-Secure FTP indicates the authentication method for a user logging into the FTP proxy from the non-secure side of the firewall.

Secure SOCKS indicates the authentication method for a user using SOCKS V5 from the secure side of the firewall.

Non-Secure SOCKS indicates the authentication method for a user using the SOCKS V5 server from the non-secure side of the firewall.

Secure HTTP indicates the authentication method for a user using the HTTP proxy from the secure side of the firewall. Valid authentication methods are permit all, deny all, firewall password, and NT logon password.

The fields for Secure and Non-Secure Administration are relevant only for firewall administrators (see 10.3, "Creating Administrators" on page 231).

You define the user as shown in Figure 279 on page 297. If you select **Firewall Password** as an authentication scheme for a service you need to set the values of the password (see Figure 280). Click on the **Password** tab. Otherwise select **OK** to create the user and close the window.

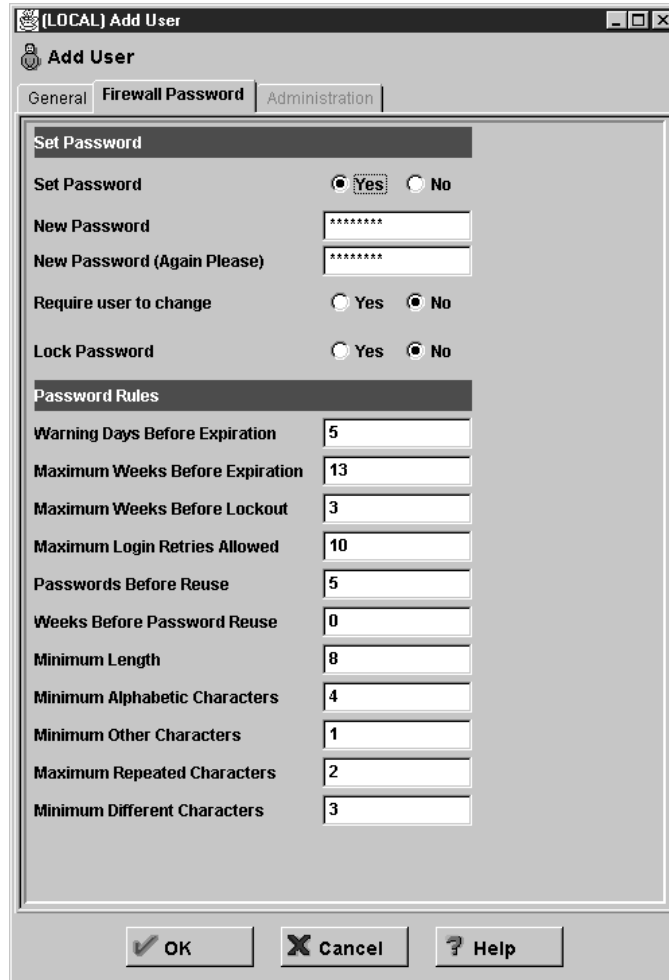


Figure 280. Password Definition Window

The options that you can specify are:

Set Password Specify Yes to assign the new password.

New Password Provide a password.

New Password (Again Please) Provide the password again, just to verify.

Require user to change Click **Yes** or **No** to indicate whether the user is required to change their password the next time they are authenticated.

Lock Password Click **Yes** or **No** to indicate whether the password is locked. This is set to Yes when the maximum number of failed logins is exceeded or when the password has not been used for the number of weeks specified in Maximum Weeks Before Lockout.

The administrator can set this field to Yes to prevent a user from using password authentication.

Warning Days Before Expiration Number of days before a password expires in which the firewall will give the user the option to change the password. The default is five days.

Maximum Weeks Before Expiration Number of weeks before the password expires. After expiration only a firewall administrator can set a new valid password. The default is 13 weeks.

Maximum Weeks Before Lockout Number of weeks in which the account is not used before it is locked out. The default is 26 weeks.

Maximum Login Retries Allowed Maximum number of failed login attempts before the password is locked. The default is 10 times.

Passwords Before Reuse Number of passwords stored in the password history list. The password cannot be changed to any password that is currently in the history list. This parameter is only valid if Weeks Before Password Reuse is zero. The default is five.

Weeks Before Password Reuse Number of weeks passwords are kept in the password history list. The password cannot be changed to any password that is currently in the history list. The default is zero weeks.

Minimum Length Minimum number of characters in a password. The default is eight characters.

Minimum Alphabetic Characters Minimum number of alphabetic characters in a password. The default is four characters.

Minimum Other Characters Minimum number of non-alphabetic characters in a password. The default is one character.

Maximum Repeated Characters Maximum number of times any single character can be repeated in the password. The default is two times.

Minimum Different Characters Minimum number of different characters in the password. The default is three characters.

After you finish entering the password and specifying the password rules, click **OK** to create the user and close the window.

If you have many users to enter, the GUI may be a rather slow method. You may want to consider using the command `fwuser` (see *IBM eNetwork Firewall for Windows NT Reference Version 3 Release 3* for parameter details). An example

batch file using the fwuser command to define proxy users follows. The user data must be provided in a separated file (see Figure 282 on page 301).

```

:: Creating Users on the Firewall
:: 06/03/98
::
:: Usage: cuser user.dat
::
@echo off
if not exist %1 echo Input file %1 not found.
if not exist %1 goto end
::
:: Generating QBASIC file
>fwcu.bas echo '
>>fwcu.bas echo ' Variables
>>fwcu.bas echo user$ = ""
>>fwcu.bas echo DIM data$(10)
>>fwcu.bas echo DIM text$(10)
>>fwcu.bas echo text$(1) = "username"
>>fwcu.bas echo text$(2) = "fullname"
>>fwcu.bas echo text$(3) = "pwdvalue"
>>fwcu.bas echo text$(4) = "secftp"
>>fwcu.bas echo text$(5) = "secauth"
>>fwcu.bas echo text$(6) = "remauth"
>>fwcu.bas echo text$(7) = "secsocks"
>>fwcu.bas echo '
>>fwcu.bas echo ' Open Files
>>fwcu.bas echo OPEN "%1" FOR INPUT AS #1
>>fwcu.bas echo OPEN "fwcu.cmd" FOR OUTPUT AS #2
>>fwcu.bas echo DO WHILE NOT EOF(1)
>>fwcu.bas echo   INPUT #1, user$
>>fwcu.bas echo   '
>>fwcu.bas echo   ' Separate Strings
>>fwcu.bas echo   j = 0
>>fwcu.bas echo   FOR i = 1 TO LEN(user$)
>>fwcu.bas echo     c$ = MID$(user$, i, 1)
>>fwcu.bas echo     IF i = LEN(user$) THEN
>>fwcu.bas echo       s$ = s$ + c$
>>fwcu.bas echo       j = j + 1
>>fwcu.bas echo       data$(j) = s$
>>fwcu.bas echo       s$ = ""
>>fwcu.bas echo     ELSEIF c$ = ":" THEN
>>fwcu.bas echo       j = j + 1
>>fwcu.bas echo       data$(j) = s$
>>fwcu.bas echo       s$ = ""
>>fwcu.bas echo     ELSE
>>fwcu.bas echo       s$ = s$ + c$
>>fwcu.bas echo     END IF
>>fwcu.bas echo   NEXT i
>>fwcu.bas echo   '
>>fwcu.bas echo   ' Construct Command
>>fwcu.bas echo   cmd$ = "fwuser cmd=add password=yes "
>>fwcu.bas echo   FOR i = 1 TO 7
>>fwcu.bas echo     cmd$ = cmd$ + text$(i) + "=" + CHR$(34) + data$(i) + CHR$(34) + " "
>>fwcu.bas echo   NEXT i
>>fwcu.bas echo   '
>>fwcu.bas echo   ' Output Command
>>fwcu.bas echo   PRINT #2, cmd$
>>fwcu.bas echo LOOP
>>fwcu.bas echo CLOSE #1
>>fwcu.bas echo CLOSE #2
>>fwcu.bas echo SYSTEM
qbasic /run fwcu.bas
fwcu
del fwcu.bas fwcu.cmd
:end

```

Figure 281. Windows NT Batch File for Creating Users

To implement this program we used a combination of standard batch files and QBASIC because on Windows NT there is no powerful scripting language available in the plain operating system. You can also use REXX or any other (scripting) language to write batch jobs using the command line interface of the IBM Firewall.

First we generate a QBASIC program. This program reads the user data input file provided as the first parameter and constructs `fwuser` commands with the appropriate data. These commands are written into an NT batch file. Finally this batch file is executed to create the users on the firewall.

The file used with the above batch file is the following:

```
pepe:Jose Garcia:raton:password:password:password:password
alicer:Alice Rogers:forgot:password:password:sdi:password
johns:John Silver:donno:password:password:deny:deny
juanp:Juan Plata:platino:password:deny:deny:deny
```

Figure 282. Sample Input File for Creating Users (user.dat)

This file contains the user ID, the user name, the password, and the authentication method for FTP from the secure side (`secftp`), telnet from the secure side (`secauth`), telnet from the non-secure side (`remauth`) and SOCKS from the secure side (`secsocks`).

We selected a few parameters of the `fwuser` command; you can modify the script to accept other parameters.

You can download this code as detailed in “How to Access Samples from This Book” on page xiii; the name of the file is `cuser.bas`.

13.1.5 Configuring the Default User

If you don't want to configure any proxy users on the firewall but you want them authenticated, then you can use the `fwdfuser` (firewall default user) user record to apply authentication methods. You can use the following authentication methods: Deny All, NT Logon, SecurID, and User-Supplied. For example, if you use the NT logon password authentication method for the HTTP proxy, all internal users, which are not explicitly defined in the firewall user database, are challenged for their NT logon password when using this services. In 13.1.6, “Configuring for NT Logon Password Authentication” on page 302 you can find details of how to set up NT logon password authentication.

Configuring the default user `fwdfuser` is the same as the configuring proxy user. Double-clicking on this row in Figure 278 on page 296 opens the following window.

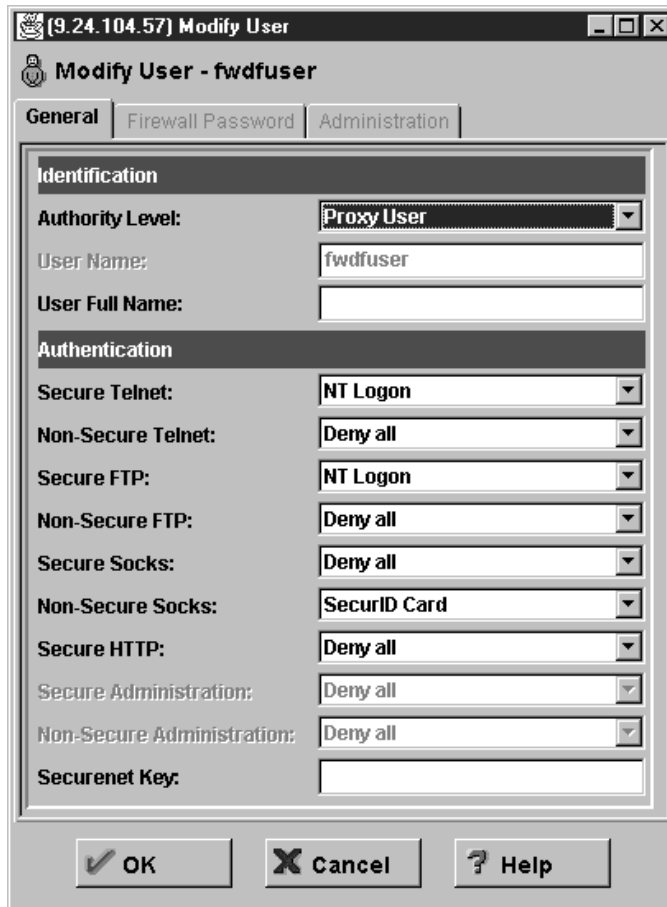


Figure 283. Default User Administration Window

With this dialog you can modify the authentication methods for firewall services that apply to all users that are not defined in the firewall user database. In our example we allow Telnet and FTP from the secure side authenticated with the NT logon password of the users, as well as using the SOCKS service from the non-secure side with the SecurID card.

13.1.6 Configuring for NT Logon Password Authentication

In order to allow users to use their NT logon password for authentication you must do the following three steps:

1. Define IP filters to allow communication between:
 - The firewall and your Primary Domain Controller (PDC)
 - The firewall and your Backup Domain Controller (BDC).
2. Enable and start **NetBIOS over TCP** services.
3. Set up the firewall as a member of your internal domain.

NT Logon Password Authentication and Security

Due to several security reasons (see 10.3.1, "The Primary Administrator Account" on page 231) we recommend that you use this authentication method only if you already have a running PDC. You must also set up very restrictive IP filter rules for the NetBIOS over TCP traffic.

13.1.6.1 Connections

To restrict the NetBIOS over TCP traffic you need to define IP filter connections (we discussed some security implications in 10.3.1, “The Primary Administrator Account” on page 231). You can use the predefined service Domain Controller Authentication. This service allows your firewall (which is a stand-alone server) to authenticate users against the Windows NT user database on your PDC or BDC.

We recommend that you set up the PDC and BDC connections as one-to-one relationships between the firewall and the server (PDC and BDC). We did this for the PDC in Figure 284.

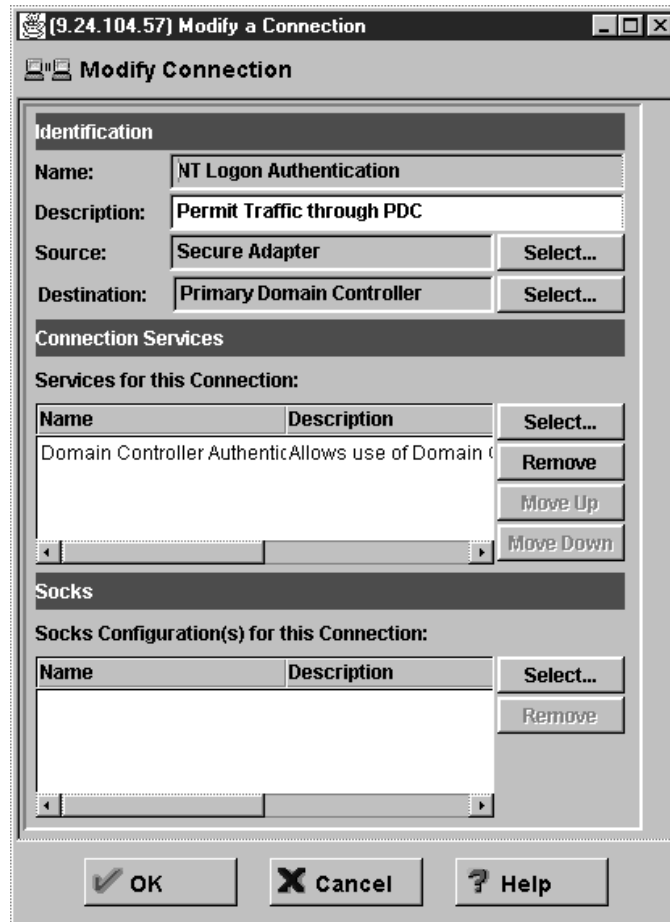


Figure 284. Connection for NT Logon Password Authentication

You find the filter rules in Figure 285.

```
# Description : Allows use of Domain Controller for user authentication
permit 9.24.104.77 255.255.255.255 9.24.104.57 255.255.255.255 udp eq 137 eq 137 secure local inbound
permit 9.24.104.57 255.255.255.255 9.24.104.77 255.255.255.255 udp eq 137 eq 137 secure local outbound
permit 9.24.104.77 255.255.255.255 9.24.104.57 255.255.255.255 udp eq 138 eq 138 secure local inbound
permit 9.24.104.57 255.255.255.255 9.24.104.77 255.255.255.255 udp eq 138 eq 138 secure local outbound
permit 9.24.104.77 255.255.255.255 9.24.104.57 255.255.255.255 tcp eq 139 gt 1023 secure local inbound
permit 9.24.104.57 255.255.255.255 9.24.104.77 255.255.255.255 tcp gt 1023 eq 139 secure local outbound
```

Figure 285. Filter Rules for PDC Authentication

The rules allow your firewall to check the passwords against the PDC user database.

13.1.6.2 Windows NT Services

In order to allow the firewall to check the NT logon passwords against the PDC, you must enable and start three Windows NT services which were disabled during firewall hardening:

- Enable and start **Workstation**
- Enable and start **TCP/IP NetBIOS Helper**
- Enable **Net Logon**

Please refer to 13.2.1, “Stopping and Starting Services” on page 307 and 13.2.2, “Disabling and Enabling Services” on page 307 to see how you can enable and automatically start services after system reboot.

These service must also be activated at your PDC and BDC.

13.1.6.3 Domain Setup

The NT logon password authentication does not work if your firewall is only a member of your internal *workgroup* because in this scenario you have no PDC with a common user database.

For the following configuration we assume that you already have a running PDC; otherwise you can't make your firewall a member of that domain.

We don't have a WINS server, therefore we use the Windows NT LMHOSTS file to do the necessary NetBIOS name lookups. The LMHOSTS is the Windows NT equivalent to hosts file for DNS name lookups. It is located in the C:\WinNT\system32\drivers\etc\ directory. You have to add LMHOSTS files to your firewall and to your PDC. Figure 286 shows the LMHOSTS file for the firewall.

```
9.24.104.77 pdc #PRE #DOM:FAIRE
9.24.104.57 ntfirewall #PRE
```

Figure 286. LMHOSTS File on the Firewall

Figure 287 shows the LMHOSTS file for the PDC.

```
9.24.104.77 pdc #PRE
9.24.104.57 ntfirewall #PRE
```

Figure 287. LMHOSTS File on the PDC

Now add your firewall to your PDC configuration. Use the Server Manager (select **Start** → **Programs** → **Administrative Tools (Common)** → **Server Manager**) on the PDC and select **Computer** → **Add to Domain** from the menu.

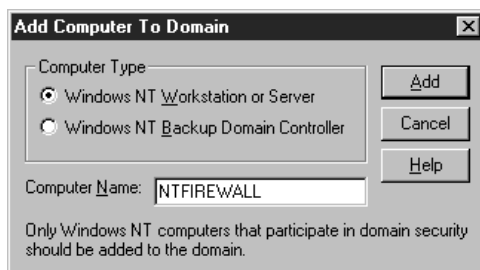


Figure 288. Add Firewall to PDC

Check the **Windows NT Workstation or Server** check box, fill in the appropriate name, and click **Add**.

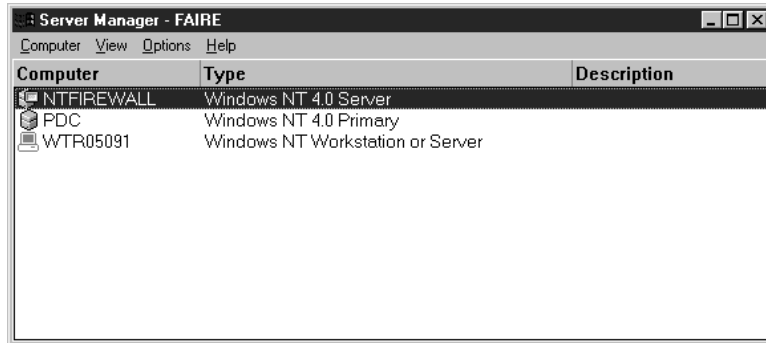


Figure 289. Server Manager

The last step is to set up your firewall as a member of your internal domain with the Network Control Panel. Click on **Start** button in the task bar and select **Settings** → **Control Panel** to open the window with all the Windows NT control panels. Now double-click the **Network** icon. As you can see in Figure 290 our firewall is a member of the workgroup FAIRE.

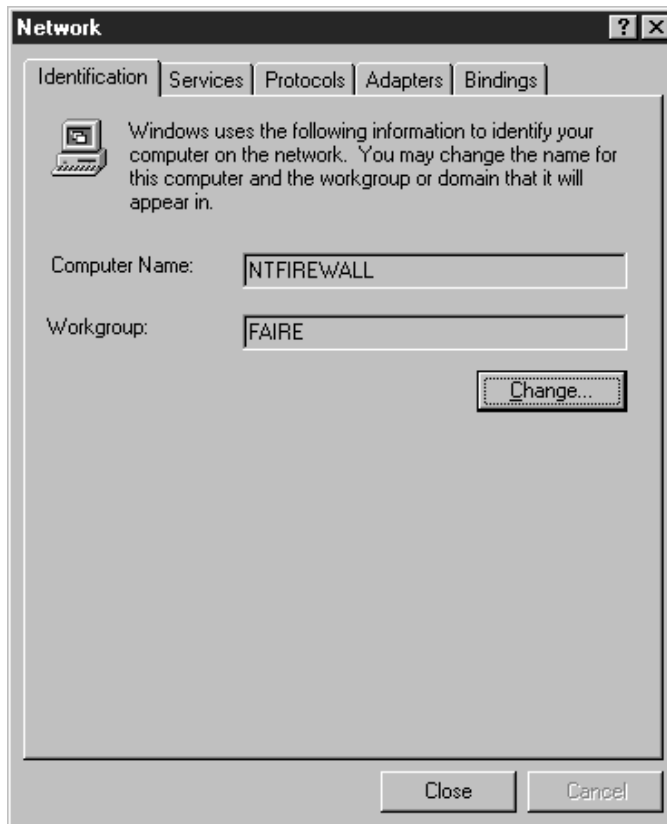


Figure 290. Network Control Panel - Identification Tab

To change this click on the **Change** button on the **Identification** tab.

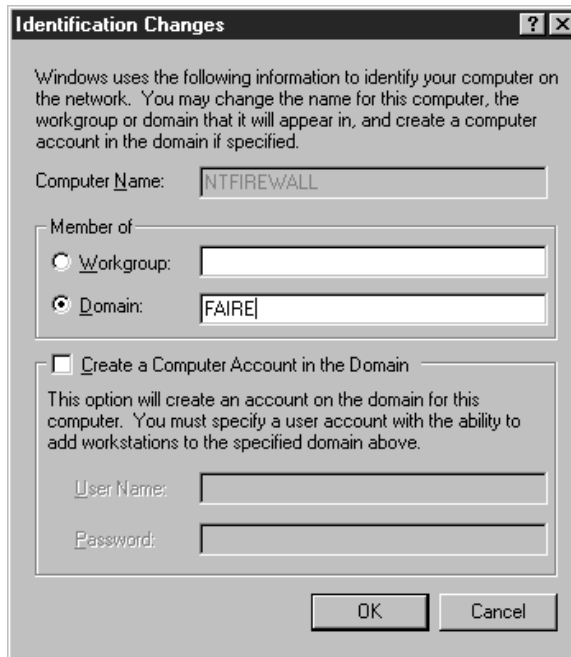


Figure 291. Identification Changes

Select the **Domain** radio button and specify your internal domain in the text field right to the button. We changed from the workgroup FAIRE to the domain FAIRE. Close both windows and restart your computer to activate the changes.

Now your firewall is a member of your internal NT Domain and you can configure users or the default user records to use the NT Logon authentication method.

13.2 Managing Proxy and SOCKS Services

The proxy and SOCKS servers from the IBM Firewall are integrated in the Windows NT environment as services. You can manage them like all other services with the Services Control Panel from Windows NT.

Five services are related to proxy and SOCKS (see Figure 292 on page 307):

- IBM Firewall Proxy FTP Daemon
- IBM Firewall Proxy HTTP Daemon
- IBM Firewall Proxy Telnet Daemon
- IBM Firewall SOCKS Monitor Service
- IBM Firewall SOCKS Server

To open the Services Control Panel select the **Start** button from the Windows NT task bar. From the following menu select **Settings** and then **Control Panel**. Now a window with all control panels appears. Double-click the **Services** icon to open the Windows NT Services Control Panel. From this window (see Figure 292 on page 307) you can start, stop, enable and disable all installed services.

13.2.1 Stopping and Starting Services

As you can see in Figure 292 all three proxy servers and the SOCKS server are configured to start automatically on system boot. You can stop or start the services with the Services Control Panel from Windows NT.

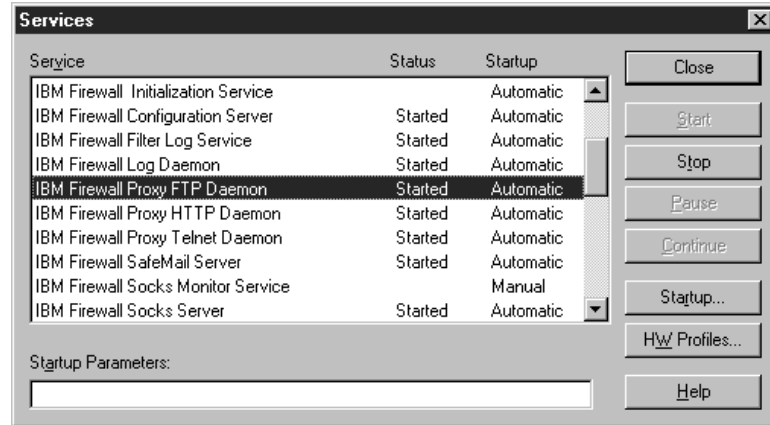


Figure 292. Services Control Panel

To stop a service, select a started service from the list and click the **Stop** button. To start a service, select a not-started service from the list and click the **Start** button. To close the window, click the **Close** button.

13.2.2 Disabling and Enabling Services

If you don't need one or more of these services because of your security policy, you should disable them. Therefore select a service and then the **Startup** button from the Services Control Panel in Figure 292 and select the **Disabled** radio button from the following window (see Figure 293).



Figure 293. Startup Type Window

To enable a service to start automatically at boot, select the **Startup** button from the Services Control Panel (see Figure 292) and select **Automatic** from Figure 293.

Click the **OK** button to close the window.

13.3 Using the Telnet and FTP Proxy

The proxy servers can operate in two different modes: the *normal* mode and the *transparent* mode. To use the normal mode, users have to be predefined in the firewall user database as proxy users. They first have to connect to the firewall, authenticate themselves, then create a second session to access the target host. With the transparent proxy server, users connect to the firewall but user authentication is not required. Instead, users enter a special user ID format. This format is `user@remote-host`. The proxy server now tries to log on to the remote host with the specified user ID. After this the user enters the password for the user ID at the remote host. There are no special accounts needed for using the transparent proxy. However, the transparent proxy is less secure, because it does not require user ID password authentication at the firewall. For this reason, the transparent proxy is only allowed from the secure network.

This section contains general information about how to configure and use the proxy servers from workstations inside and outside your secure network.

Access to the Firewall

Neither the FTP proxy nor the Telnet proxy allows access to the firewall itself.

Passive FTP

The actual implementation of the FTP proxy does not allow passive FTP. You can use passive FTP with the HTTP proxy.

13.3.1 Configuring Proxy Services

To use the proxy services for Telnet and FTP provided by the IBM Firewall with user-specific authentication methods you have to do two things:

1. Add users to the firewall user database and specify their allowed services and authentication methods. This is explained in detail in 13.1, "User Administration" on page 294.
2. Add connections for the desired proxy services.

If you don't want to configure all users on the firewall, you can define the firewall default user `fwdfuser` (see 13.1.5, "Configuring the Default User" on page 301), but you still have to add connections to the firewall.

13.3.1.1 Outbound Connections

Before the proxy server can work properly, some connections must be set up to allow the desired FTP and Telnet traffic.

Figure 294 on page 309 and Figure 295 on page 310 show the required connections for the use of outbound FTP and Telnet using the firewall proxy servers. As you can see, the connection is divided into two parts: one from the secure network to the secure adapter of the firewall, allowing clients to connect to the firewall proxy server, and one from the non-secure adapter to the non-secure network, allowing the firewall proxy server to connect to the target hosts.

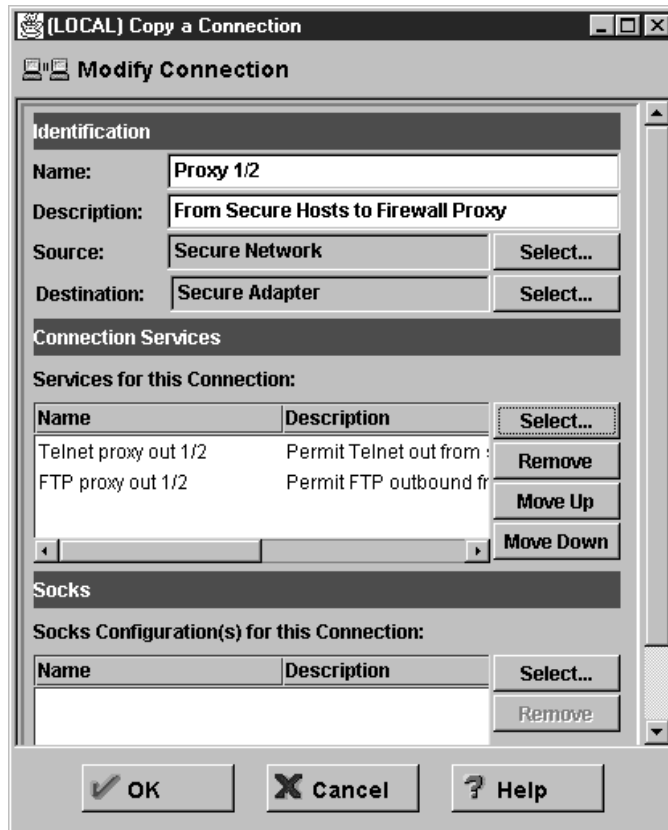


Figure 294. Proxy Connection on Secure Network

You can select the source and destination for the connections from the list of network objects (see 8.2, “Rule Base Design” on page 151) by clicking the corresponding **Select** button. You can also select the desired services for the connections from a list of predefined services (see 8.2, “Rule Base Design” on page 151).

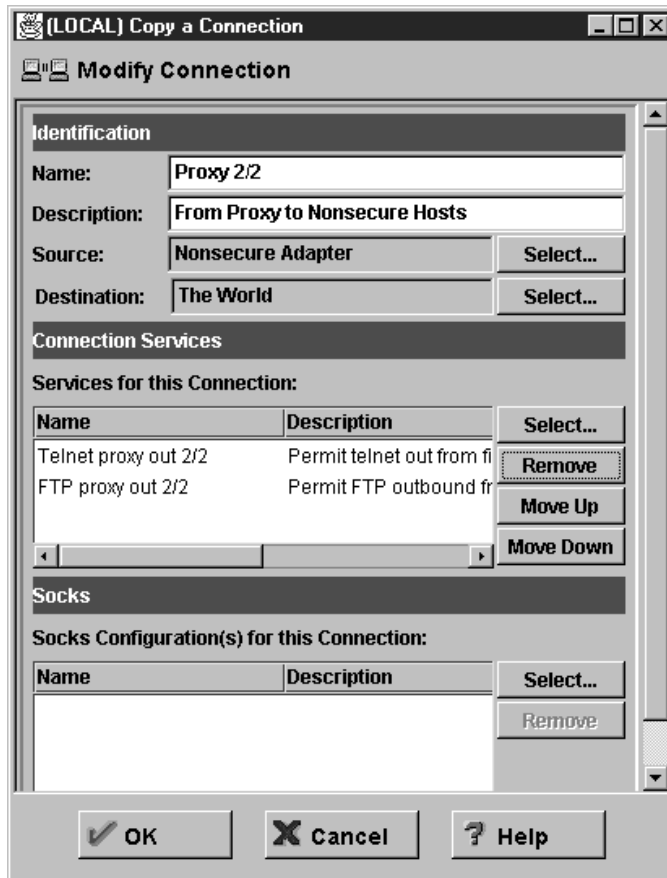


Figure 295. Proxy Connection on Non-secure Network

In this example all hosts from the secure network are allowed to establish Telnet and FTP connections to hosts in the non-secure network using the firewall proxy server.

13.3.1.2 Inbound Connections

The definition of inbound connections from non-secure hosts to inside secure servers are also possible but you should keep some security considerations in mind:

- Define the sources as hosts where possible. This allows only a few hosts to connect to your servers, not whole networks.
- Define the destinations also as hosts.
- Define the connections only for the needed services. The reason for this is to open as few as possible ports for inbound connections.
- Use one-time password such as SecurID Card (see 21.5, "SecurID Authentication" on page 563) and encryption techniques to secure the login procedure and the traffic through the external network.

13.3.1.3 Filter Rules

There is no point in setting up a proxy service if users can bypass it by routing through the firewall. As discussed in Chapter 8, “IBM Firewall Rule Base” on page 143 you are not allowed to edit the filter rules file manually because it will be generated each time you *regenerate the connection rules* from the configuration files. But you can look at the filter rules file to review the generated rules as we have done in Figure 296. We have already shown the filter rules for the Telnet and FTP proxy server (see 9.3, “Telnet” on page 170 and 9.4, “FTP: File Transfer Protocol” on page 175), but we show them again here for completeness.

```
# Description : Permit Telnet out from secure network to firewall
permit 9.24.104.0 255.255.255.0 9.24.104.57 255.255.255.255 tcp gt 1023 eq 23 secure local inbound
permit 9.24.104.57 255.255.255.255 9.24.104.0 255.255.255.0 tcp/ack eq 23 gt 1023 secure local outbound
# Description : Permit telnet out from firewall to non-secure network
permit 150.53.104.56 255.255.255.255 0 0 tcp gt 1023 eq 23 non-secure local outbound
permit 0 0 150.53.104.56 255.255.255.255 tcp/ack eq 23 gt 1023 non-secure local inbound

# Description : Permit FTP outbound from secure network to firewall
permit 9.24.104.0 255.255.255.0 9.24.104.57 255.255.255.255 tcp gt 1023 eq 21 secure local inbound
permit 9.24.104.57 255.255.255.255 9.24.104.0 255.255.255.0 tcp/ack eq 21 gt 1023 secure local outbound
permit 9.24.104.57 255.255.255.255 9.24.104.0 255.255.255.0 tcp eq 20 gt 1023 secure local outbound
permit 9.24.104.0 255.255.255.0 9.24.104.57 255.255.255.255 tcp/ack gt 1023 eq 20 secure local inbound
permit 9.24.104.0 255.255.255.0 9.24.104.57 255.255.255.255 tcp gt 1023 gt 1023 secure local inbound
permit 9.24.104.57 255.255.255.255 9.24.104.0 255.255.255.0 tcp/ack gt 1023 gt 1023 secure local outbound
# Description : Permit FTP outbound from firewall to non-secure network
permit 150.53.104.56 255.255.255.255 0 0 tcp gt 1023 eq 21 non-secure local outbound
permit 0 0 150.53.104.56 255.255.255.255 tcp/ack eq 21 gt 1023 non-secure local inbound
permit 0 0 150.53.104.56 255.255.255.255 tcp eq 20 gt 1023 non-secure local inbound
permit 150.53.104.56 255.255.255.255 0 0 tcp/ack gt 1023 eq 20 non-secure local outbound
permit 150.53.104.56 255.255.255.255 0 0 tcp gt 1023 gt 1023 non-secure local outbound
permit 0 0 150.53.104.56 255.255.255.255 tcp/ack gt 1023 gt 1023 non-secure local inbound
```

Figure 296. Filter Rules for Telnet and FTP Using the Proxy Server

The first pair of rules allows telnet access to the firewall from a node inside the secure network (you need this to get to the proxy server in the first place). The second pair of rules allows Telnet sessions started from the firewall to the non-secure network.

These rules only allow access from Telnet clients inside the secure network to servers outside it. As we explained previously, there are serious security implications if you provide Telnet access from the non-secure network to the secure network (see 1.2, “What Does “Security” Mean?” on page 3).

Next are the rules for the ftp connections. You first have rules to connect to the firewall and then rules to allow the firewall to connect to the final destination. This example includes rules for *normal mode* and for *passive mode* ftp. See 9.4, “FTP: File Transfer Protocol” on page 175 for the differences.

13.3.2 Example Proxy Connections

In order to use the Telnet proxy, the user in the secure network enters the `telnet` command to connect to the firewall (9.24.104.57). Figure 297 on page 312 shows a typical session.

```

C:\>telnet 9.24.104.57
Proxy Telnet (ntfirewall)
login:sven
Password:****
sven has been authenticated.
Enter desired host: 150.53.104.12
Trying...
Connected to 150.53.104.12.
Escape character is '['.

HP-UX ns1 A.09.05 A 9000/715 (ttyq1)

login: root
Password:

Value of TERM has been set to "aixterm".
WARNING: YOU ARE SUPERUSER !!
/ >

```

Figure 297. Outbound Telnet Proxy Session

After the users are authenticated by the IBM Firewall, they again Telnet to the final destination. Notice also that this example demonstrates the most common failing of security measures. The proxy server is doing a good job of protecting the secure network, but the target machine (150.53.104.12) has been compromised because the user sven connected to it with the root user ID. Under Telnet, the password is sent in clear text, so anyone in the non-secure network could have captured it.

Figure 298 is an example of a user session using the FTP proxy. The user uses the ftp command to connect to the firewall (9.24.104.57). Once there, they are authenticated, and then they use quote site in order to reach their final destination.

```

C:\>ftp 9.24.104.57
Connected to ntfirewall.
220 ntfirewall FTP GATEWAY (Version 1.0) ready.
User (ntfirewall:(none)): sven
331 Password required for sven.
Password:
230 To specify destination, type "quote site remote.host.com"
ftp> quote site 150.53.104.8
220 www Microsoft FTP Service (Version 2.0).
ftp> ls
200 PORT command successful.
530 Please login with USER and PASS.
ftp> user anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password:
230-Hello, welcome to ftp.faire.com!
230 Anonymous user logged in.
ftp> dir
200 PORT command successful.
150 Opening ASCII mode data connection for /bin/ls.
----- 1 owner   group           263 Aug  9 1996 file1.dat
----- 1 owner   group       6816726 Jan 29 20:05 file2.dat
----- 1 owner   group        1236 Jan 29 20:25 image1.gif
226 Transfer complete.
229 bytes received in 0,03 seconds (7,63 Kbytes/sec)
ftp> quit
Goodbye.

C:\>

```

Figure 298. Outbound FTP Proxy Session

The next example shows you an inbound connection using the normal ftp proxy on the firewall.

```
C:\>ftp 150.53.104.56
Connected to 150.53.104.56.
220 ntfirewall FTP GATEWAY (Version 1.0) ready.
User (150.53.104.56:(none)): sven
331 Password required for sven.
Password:
230 To specify destination, type "quote site remote.host.com"
ftp> quote site 9.24.104.70
220 FTP Software, Inc. Win32 FTP Server 2.0.0.213 ready.
ftp> user itso
331 Password required for itso.
Password:
230 User itso logged in.
ftp> dir
200 PORT command successful.
150 Opening ASCII mode data connection for /bin/ls.
02-25-98 12:54PM <DIR> ..
02-10-98 06:51PM 24669421 CLIENT.ZIP
02-20-98 00:01PM 4096 FWFILTERS.CFG
226 File transfer complete.
143 bytes received in 0,09 seconds (7,74 Kbytes/sec)
ftp> quit
Goodbye.

C:\>
```

Figure 299. Inbound FTP Proxy Session

The user on the outside host 150.53.104.12 opens an ftp connection to the non-secure adapter of the firewall (150.53.104.56). After the authentication the `quote site` command is used to connect to the final ftp server (9.24.104.70). Without additional security the outbound session looks like the inbound session. We recommend additional security for inbound sessions; see 13.3.1.2, “Inbound Connections” on page 310.

13.4 Using the Transparent Proxy

This section contains general information about how to configure and use the transparent proxy servers from workstations inside your secure network.

13.4.1 Configuring Transparent Proxy Services

Once the proxy servers are set up, it's quite easy to enable them for transparent mode. From the IBM Firewall main navigation tree, select **System Administration**, and then **Security Policy**, now you can choose to enable the transparent Telnet and/or FTP proxy. We enabled both in Figure 300 on page 314.

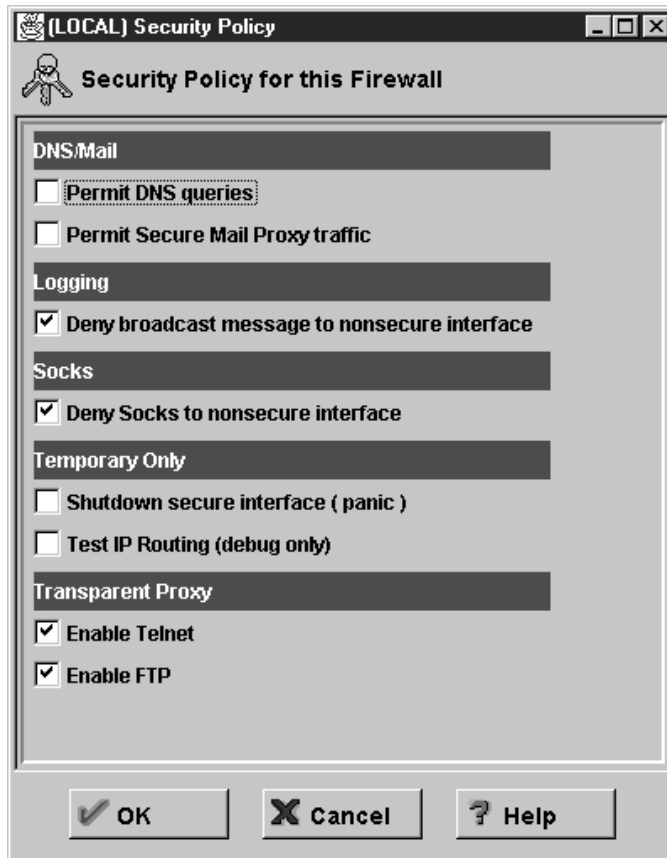


Figure 300. Enable Transparent Telnet and FTP Proxy

Setting up the connections for the transparent proxy is the same as for the normal proxy. See 13.3.1.1, "Outbound Connections" on page 308 for details.

13.4.2 Example Transparent Proxy Connections

Figure 301 is an example of a transparent Telnet proxy session. The user only starts a telnet session to the firewall and enters `root@150.53.104.12` at the login prompt. After this only the password for root at 150.53.104.12 has to be entered.

```
C:\>telnet ntfirewall
Proxy Telnet (ntfirewall)
login:root@150.53.104.12
Trying...
Connected to 150.53.104.12.
Escape character is '^['.

HP-UX ns1 A.09.05 A 9000/715 (ttyq1)

login:Password:

Value of TERM has been set to "aixterm".
WARNING: YOU ARE SUPERUSER !!
/ >
```

Figure 301. Transparent Telnet Proxy Session

Figure 302 on page 315 is an example of a user session using the transparent FTP proxy. The user only starts an ftp session to the firewall and enters

anonymous@150.53.104.8 at the login prompt. After this only the password for anonymous at 150.53.104.8 has to be entered.

```
C:\>ftp ntfirewall
Connected to ntfirewall.
220 ntfirewall FTP GATEWAY (Version 1.0) ready.
User (ntfirewall:(none)): anonymous@150.53.104.8
331 Anonymous access allowed, send identity (e-mail name) as password.
Password:
230-Hello, welcome to ftp.faire.com!
230 Anonymous user logged in.
ftp> quit
Goodbye.

C:\>
```

Figure 302. Transparent FTP Proxy Session

13.5 Idle Proxy Connections

In this section, we discuss idle proxy connections. The purpose of a proxy server is to provide access to outside networks for internal users (or vice versa). There is no reason to allow users to establish these connections and then do nothing. Idle users tie up resources on the firewall. A process disconnects all non-interactive sessions to the IBM Firewall after a specific period of idle time.

13.5.1 Configuring Idle Proxy Connections

Both FTP and Telnet have timeout values for logging in and idle waits.

By default, there must be session activity at least once every 60 seconds during login and user authentication. This is known as the *loginTimeout*. Once the login has completed successfully, there must be activity on the session at least once every 7200 seconds or the session is disconnected.

You can override these defaults by creating an `fwTimeout.cfg` file in the configuration directory (`C:\Program Files\IBM\Firewall\config\`) and by specifying new timeout values in seconds. The `fwTimeout.cfg` file should have the following format. All values are specified in seconds.

```
telnet
proxyTimeout=7200
loginTimeout=60

ftp
proxyTimeout=7200
loginTimeout=60
```

Figure 303. Proxy Timeout Definition File

You cannot set up special timeout values per user ID.

13.6 Using the HTTP Proxy

The IBM Firewall provides an HTTP proxy server which handles browser requests efficiently. Users can access useful information on the Internet, without compromising the security of their secure networks and without altering their client environment to implement HTTP proxy.

The user needs to change the proxy pointer on the configuration page of their browser to point to the IBM Firewall and the proper port.

Content and Caching?

The HTTP proxy is not a server. The end user cannot load files of the proxy or put files on the proxy. Also, it is not a caching proxy. Nothing is stored on the firewall on behalf of an HTTP request.

The HTTP proxy can only be used for outbound connections. Besides the HTTP protocol, the proxy also supports FTP, Gopher, WAIS and HTTPS.

13.6.1 Authentication Levels

Table 24 shows all valid choices for user authentication levels from the authentication pop-up menu in Figure 304 on page 317.

Level	Description
None	If you select None the proxy does not check browser requests and does not prompt for any user id and password.
All	If you select All then all browsers will be sent the <i>407 proxy authenticate response</i> to indicate that the browser should prompt the user for a user ID and password. If the browser does not support the 407 proxy authenticate response, the 407 error page displays indicating this. If the browser supports it, the user ID and password prompt will be displayed. We have tested this method successfully with the Netscape Navigator version 3.0, Netscape Communicator version 4.04, and Microsoft Internet Explorer version 4.
HTTP 1.1+	The selection HTTP 1.1+ is used as a migration aid. It will only send back a 407 proxy authenticate response to tell the browser to issue a user ID/password prompt, to a client browser that identifies itself as an HTTP/1.1 browser. You can set a switch in Internet Explorer 4.0 so that it will broadcast requests with the HTTP/1.1 identifier. Netscape and others identify themselves as HTTP/1.0 requests.

13.6.2 Configuring HTTP Proxy Services

If you provide the HTTP proxy for your users you should make sure that DNS queries are permitted from the firewall. Otherwise, the proxy cannot resolve any external hostnames in the browser requests. You could do that by enabling **Permit DNS Queries** in the Security Policy window (see Figure 300 on page 314) but we recommend you set up your own more restricted connections for DNS (see 9.7, "DNS: Domain Name Server" on page 188 and Chapter 11, "Domain Name Service (DNS)" on page 247 for details).

The configuration parameters of the proxy server can be modified using the GUI. Figure 304 on page 317 shows the HTTP proxy configuration window. To open the window click on **HTTP** in the navigation tree.

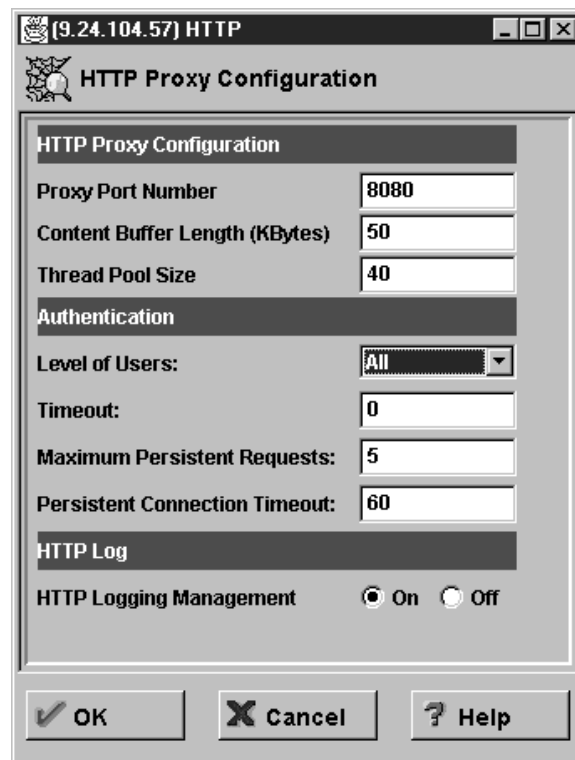


Figure 304. HTTP Proxy Configuration

The parameters that you can modify are:

Proxy Port Number Use this parameter to specify the port number the proxy should listen to for requests. If you change the port number, you must configure your filters to allow or disallow flow through the ports. Port numbers less than 1024 are reserved for TCP/IP applications. Common ports used for proxy Web servers are 8080 and 8088.

If you change this, the port number must also be changed in the Services that are set up for this configuration and of course in all clients using the HTTP proxy (see 13.6.3, “Example HTTP Proxy Connection” on page 321).

Content Buffer Length (KB) Use this parameter to set the size of the buffer for dynamic data generated by a server. Dynamic data is output from CGI programs, server-side includes, and API programs. It is data that does not come from a proxy.

Specify the value in kilobytes (KB). The default is 50 KB.

Thread Pool Size Use this parameter to set the fixed number of threads that you want to have active at one time. You can understand threads as lightweight processes within processes. The advantage is the faster switching between several threads as between processes. If the maximum number of threads is reached, the proxy holds new requests until another request finishes and threads become available. Generally, the more power a machine has, the higher the value you should use for

this parameter. If a machine starts to spend too much time on overhead tasks, such as swapping memory, try reducing this value. Specify a whole number such as 60. The default is 40.

Level of Users This parameter tells the proxy what level is used for user authentication. Specify the value as either **None**, **All**, or **HTTP 1.1+**. The default is none. See Table 24 on page 316 for details.

Timeout This parameter tells the proxy how much time to wait for a client request before requiring the user to reauthenticate himself or herself. A user is authenticated from the specific IP address and user ID given at the time of the original authentication for this period of idle time. Specify the time in minutes. The default is 60 minutes.

As long as the user is actively browsing, this time window will not expire.

Maximum Persistent Requests This parameter indicates the maximum number of requests that a proxy can receive on an HTTP/1.1 persistent connection. This is a performance tool that directly impacts the authentication timeout. While in a persistent session, no test of the authentication of a user is done until the persistent session ends. Specify the value as a whole number, for example 25. The default is 5.

Persistent Connection Timeout This parameter indicates the time in seconds to keep an HTTP/1.1 persistent connection with a client browser once an HTTP/1.1-compliant browser starts a session with the proxy. This is a performance tool that directly impacts the authentication time out. While in a persistent session, no test of the authentication of a user is done until the persistent session ends. Specify the time in minutes. The default is 60.

HTTP Logging Management This parameter tells the proxy to log startup/shutdown and all proxy requests to the firewall log. It uses the LOG_NOTICE level of logging. Set this to on if you wish to monitor HTTP request activity. Events are logged in the C:\Program Files\IBM\Firewall\log\local4.log in the Windows NT install tree. See Chapter 16, "Logging, Monitoring, and Reporting" on page 409 for details.

If you change any of these settings you must restart the IBM Firewall Proxy HTTP Daemon service. The Configuration Client asks you to do this for you.

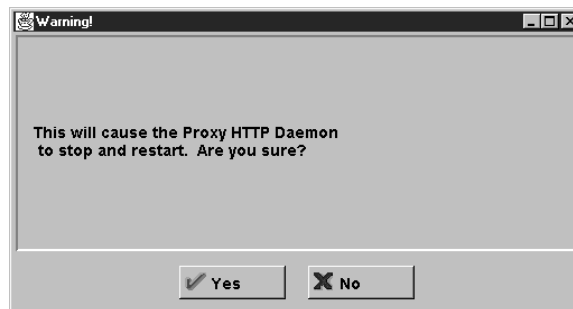


Figure 305. HTTP Proxy Restart

13.6.2.1 Connections

Before the HTTP proxy server can work properly, some connections must be set up to allow the packet traffic. The rules for an HTTP proxy server have been shown in 9.9, "HTTP - World Wide Web Sessions" on page 192. However, as we mentioned earlier, the HTTP proxy server also support FTP, Gopher, WAIS and HTTPS. So if such services are allowed, you need to enable them on the packet filter as well. The filtering rules for these services are described in Chapter 9, "Examples of Rules for Specific Services" on page 161. Note that when these additional services are used, they are encapsulated in the HTTP connection between the internal client and the proxy.

Figure 306 and Figure 307 on page 320 show the HTTP proxy connections required in the secure and non-secure network respectively.



Figure 306. HTTP Proxy Connection on Secure Network

For the connection from the secure hosts to the firewall you only need one service because other protocols supported though the HTTP proxy are tunneled through HTTP.

But for the outside connection you have to specify all desired protocols. In this example we have specified FTP, HTTP, HTTPS, Gopher, and WAIS. The last two protocols are not shown in Figure 307 on page 320.

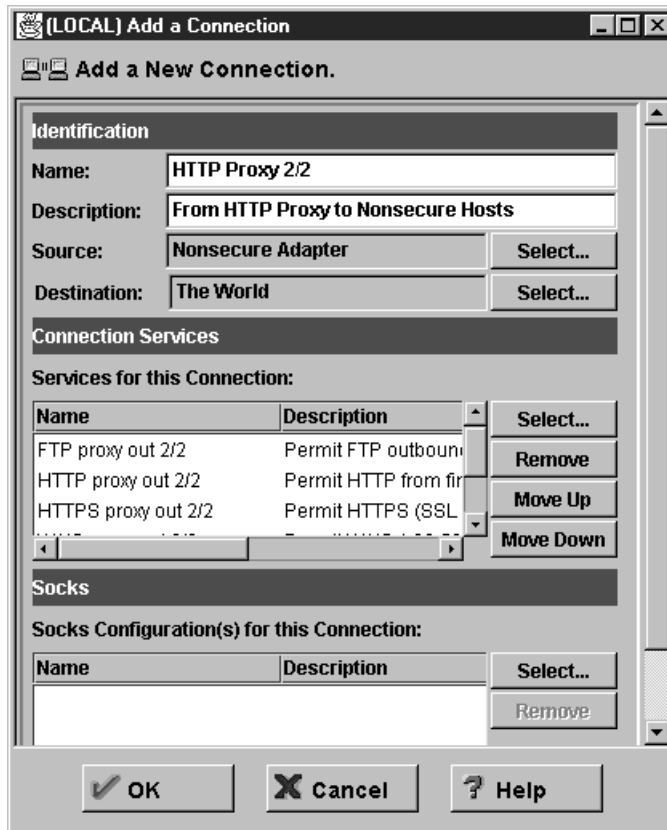


Figure 307. HTTP Proxy Connection on Non-secure Network

13.6.2.2 Filter Rules

We take now a look at the filtering rules generated for the connections in 13.6.2.1, "Connections" on page 319.

```

# Description : Permit HTTP (port 8080) from secure network to the firewall
permit 9.24.104.0 255.255.255.0 9.24.104.57 255.255.255.255 tcp gt 1023 eq 8080 secure local inbound
permit 9.24.104.57 255.255.255.255 9.24.104.0 255.255.255.0 tcp/ack eq 8080 gt 1023 secure local outbound

# Description : Permit FTP outbound from firewall to non-secure network
permit 150.53.104.56 255.255.255.255 0 0 tcp gt 1023 eq 21 non-secure local outbound
permit 0 0 150.53.104.56 255.255.255.255 tcp/ack eq 21 gt 1023 non-secure local inbound
permit 0 0 150.53.104.56 255.255.255.255 tcp eq 20 gt 1023 non-secure local inbound
permit 150.53.104.56 255.255.255.255 0 0 tcp/ack gt 1023 eq 20 non-secure local outbound
permit 150.53.104.56 255.255.255.255 0 0 tcp gt 1023 gt 1023 non-secure local outbound
permit 0 0 150.53.104.56 255.255.255.255 tcp/ack gt 1023 gt 1023 non-secure local inbound

# Description : Permit HTTP from firewall to non-secure network
permit 150.53.104.56 255.255.255.255 0 0 tcp gt 1023 eq 80 non-secure local outbound
permit 0 0 150.53.104.56 255.255.255.255 tcp/ack eq 80 gt 1023 non-secure local inbound

# Description : Permit HTTPS (SSL tunnel) from firewall to non-secure network
permit 150.53.104.56 255.255.255.255 0 0 tcp gt 1023 eq 443 non-secure local outbound
permit 0 0 150.53.104.56 255.255.255.255 tcp/ack eq 443 gt 1023 non-secure local inbound

# Description : Permit gopher from firewall to non-secure network
permit 150.53.104.56 255.255.255.255 0 0 tcp gt 1023 eq 70 non-secure local outbound
permit 0 0 150.53.104.56 255.255.255.255 tcp/ack eq 70 gt 1023 non-secure local inbound

# Description : Permit WAIS (z39.50) from firewall to non-secure network
permit 150.53.104.56 255.255.255.255 0 0 tcp gt 1023 eq 210 non-secure local outbound
permit 0 0 150.53.104.56 255.255.255.255 tcp/ack eq 210 gt 1023 non-secure local inbound

```

Figure 308. Filter Rules for FTP, HTTP, HTTPS, Gopher, and WAIS Using the HTTP Proxy

The first pair of rules allows any access to port 8080 of the firewall from any node inside the secure network. As explained earlier, HTTPS, Gopher, FTP, and WAIS are tunneled through HTTP on port 8080. For this reason you don't need filter rules for these protocols through the firewall. But from the firewall to the destination hosts you have to specify filter rules for all desired protocols. In Figure 308 you see rules for normal and passive mode FTP, HTTP, HTTPS, Gopher and WAIS.

13.6.3 Example HTTP Proxy Connection

We present you two common connection examples for the HTTP proxy. One is a direct connection from your internal clients to the firewall HTTP proxy. The other example explains proxy chaining. Both have some advantages and disadvantages. They are discussed in the sections.

13.6.3.1 Browser Connection

In order to use the HTTP proxy, all users have to change their proxy pointer in the browser to point to the IBM Firewall and the proper port.

To open the Preferences window from Netscape Navigator Version 4, select **Edit** from the menu and then **Preferences**. From the Category navigation tree select **Advanced** and click on **Proxies**. Figure 309 on page 322 shows the Preferences window from the Navigator.

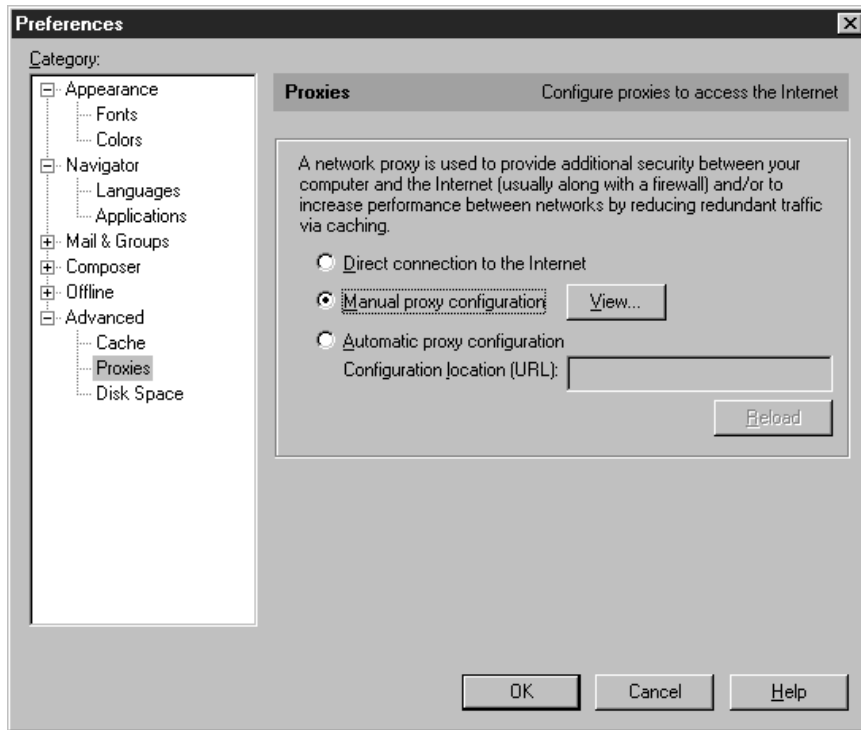


Figure 309. Netscape Navigator Proxy Preferences Window

Now switch the radio button to **Manual proxy configuration** and click on the **View** button right from this choice. You should now see the Manual Proxy Configuration window. Figure 310 shows a typical configuration.

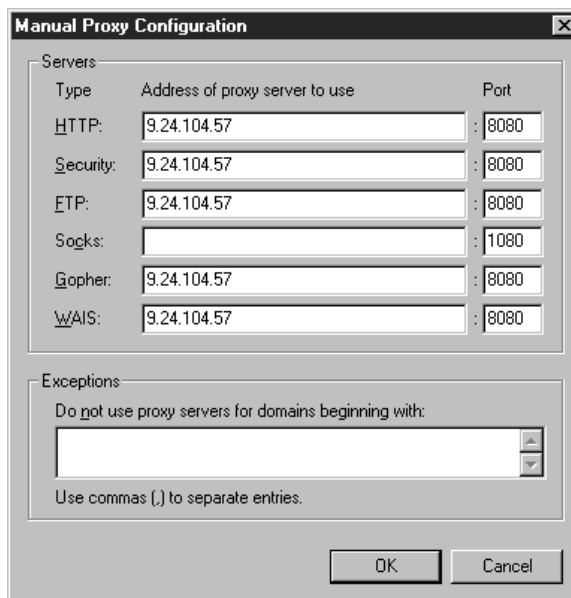


Figure 310. HTTP Proxy Configuration for Netscape Navigator

As you can see, the IBM Firewall is the proxy for HTTP, Security (HTTPS), FTP, Gopher, and WAIS. The browser channels all requests for other protocols in HTTP to the firewall port 8080. The firewall connects to the protocol-specific port on the destination host.

Let's assume your browser is configured as above and all users must be authenticated on the proxy (see 13.6.1, "Authentication Levels" on page 316). If you open the URL <http://150.53.104.8/>, the browser will respond with the following window.



Figure 311. 407 Proxy Authenticate Response Window

You have to enter your user ID and password according to the definitions made for you for the HTTP proxy in the proxy user definition.

If you implement this scenario, you can use all protocols that are natively supported by the IBM Firewall without the need for any additional software in the intranet. But this proxy doesn't cache any pages like other Web proxies (see 21.3, "IBM Web Traffic Express" on page 529). Therefore, each request retrieves all pages and files from their original location. This increases your network traffic from the Internet.

13.6.3.2 Chaining Proxy Servers

If you want to chain proxy servers, the IBM Firewall must be the last proxy server in this chain. The firewall itself cannot connect to any further proxy server, for example the caching proxy of your Internet Service Provider. See Figure 312 on page 324 for an example scenario.

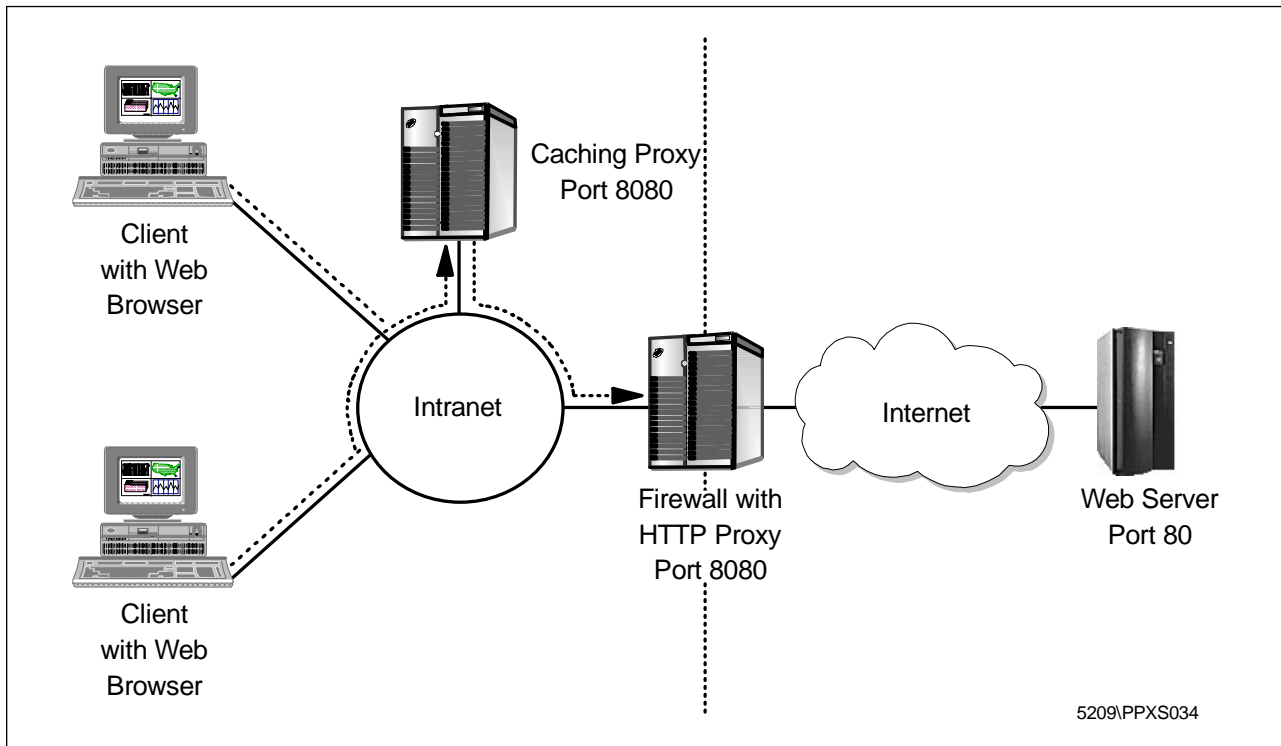


Figure 312. Proxy Chaining Example

As you see in this example, we are using an internal Web proxy, for example Web Traffic Express from IBM (see 21.3, "IBM Web Traffic Express" on page 529), to cache Web pages and FTP downloads. All internal clients point to this proxy. And this proxy points to the firewall HTTP proxy. This decreases your Internet traffic but your internal clients can only use the protocols supported by the internal Web proxy.

13.7 Using the SOCKS Server

SOCKS is an Internet standard for circuit-level gateways. You use the SOCKS server for address translation if your application uses TCP, such as Web browsers, FTP, or Telnet applications. SOCKS can help you access the Internet, while hiding your internal IP addresses.

The SOCKS server can intercept all outbound TCP/IP requests that would cross between your network and the Internet. The SOCKS server provides a remote application program interface so that the functions executed by client programs in secure domains are piped through secure servers at the firewall, hiding the client's IP address. Access is controlled by filters that are associated with the SOCKS rules.

The SOCKS server is similar to the proxy server. But while the proxy server actually performs the TCP/IP function at the firewall, the SOCKS server just identifies the user and redirects the function through the firewall. The actual TCP/IP function is performed at the client workstation, not at the firewall. This saves processing in the firewall. The users in the secure network can use the many TCP/IP products that support the SOCKS standard.

The IBM Firewall provides the SOCKS server Version 5 protocol, which enables clients inside and outside the secure network to pass an authentication stage before accessing applications in the other network. The SOCKS protocol does not encrypt the data stream between the client and the server. Especially for inbound connections you should use additional encryption tools to protect the authentication and the data traffic.

The SOCKS V5 server also provides an authenticated generic proxy and the ability to proxy some (UDP-based) streaming audio and video protocols.

The SOCKS V5 server supports the following protocols:

- Archie
- Finger
- FTP
- Gopher
- HTTP
- HTTP Proxy
- News
- SNMP
- Telnet
- TFTP
- RealAudio
- RealPlayer
- Whois
- X-Windows

Like the proxy servers, the SOCKS daemon runs as a Windows NT service automatically starting at boot time. See 13.2, “Managing Proxy and SOCKS Services” on page 306 for details on managing the service. In addition, a Watch Agent is provided to allow monitoring of the server. You can start this service manually if you wish (see 16.5.2, “Starting the SOCKS Traffic Monitor” on page 444 for details).

13.7.1 User Authentication Modes

The IBM Firewall provides a smooth migration path in the form of three authentication modes so that customers can continue to use installed SOCKS V4 clients as they introduce SOCKS V5 clients.

Table 25 (Page 1 of 2). SOCKS Authentication Modes

Mode	Description	Tag
permissive	The most permissive profile does not enable outbound authentication and permits any user, whether using a Version 4 or Version 5 client to connect. In this scenario inbound connections are denied.	1

Mode	Description	Tag
intermediate	The migration profile allows SOCKS V4 users to pass unauthenticated, but requires SOCKS V5 users to authenticate. Inbound SOCKS V4 connections are denied and inbound SOCKS V5 connections are required to authenticate. This is the default profile.	2
strict	The most secure profile requires that all users use SOCKS V5 clients and provide valid authentications.	3

SOCKS V5 clients can be authenticated by any of the supported authentication schemes (see 13.1.3, "User Authentication Schemes" on page 295). However, clients must support the Challenge-Response Authentication Method (CRAM) protocol in order to use strong authentication. Otherwise only user ID and password schemes can be applied.

You cannot define the authentication mode with the Configuration Client. You have to edit a file (C:\Program Files\IBM\Firewall\config\explode.cfg). Use the tag shown in Table 25 on page 325 to indicate your desired profile. The default profile is intermediate. Figure 313 shows the file explode.cfg.

```
# This file controls certain aspects of how the IBM Firewall
# constructs (explodes) configuration files.
#
# socks5profile
# Determines the "authentication profile" to be used for building
# the socks5 config file (socks5.conf)
# 1 = permissive mode, no authentication is performed
# 2 = intermediate (migration) mode, socks5 clients must provide
#   credentials, but socks4 clients may pass
# 3 = strict mode, all clients must authenticate (socks4 is
#   disallowed)
socks5profile=2
```

Figure 313. SOCKS User Authentication Profile explode.cfg

Be sure to regenerate the connection rules after editing this file to reflect authentication changes.

13.7.2 Configuring SOCKS Services

When the firewall is installed, the SOCKS server is enabled, but there are no connections defined. For SOCKS clients to use the SOCKS server, you must first configure SOCKS using the configuration client.

To set up SOCKS services on the IBM Firewall, you need to:

1. Build a connection from the secure network to the IBM Firewall SOCKS server.
2. Build a connection from the IBM Firewall SOCKS server to the non-secure network.
3. Specify SOCKS server configuration for connections between the secure network and the non-secure network.

The first two steps are similar to configuring proxy connections. But the third step is unique for SOCKS connections. All SOCKS server configuration entries are placed in the file C:\Program Files\IBM\Firewall\config\socks5.conf.

Currently only a subset of SOCKS V5 configuration is supported via the Configuration Client. The SOCKS V5 server supports a full SOCKS V5 configuration via file editing (see 13.7.3, “Advanced Configuration” on page 331 for details).

13.7.2.1 Connections

First you create a connection from the secure network to the IBM Firewall SOCKS server. Figure 314 shows the GUI window to build this connection.



Figure 314. Connection from Secure Network to SOCKS Server

In this example, the entire secure network is allowed to connect to the firewall SOCKS server on port 1080.

As the next step you create a connection from the IBM Firewall SOCKS server to the non-secure network. Figure 315 on page 328 shows the GUI window to build this SOCKS outbound connection.

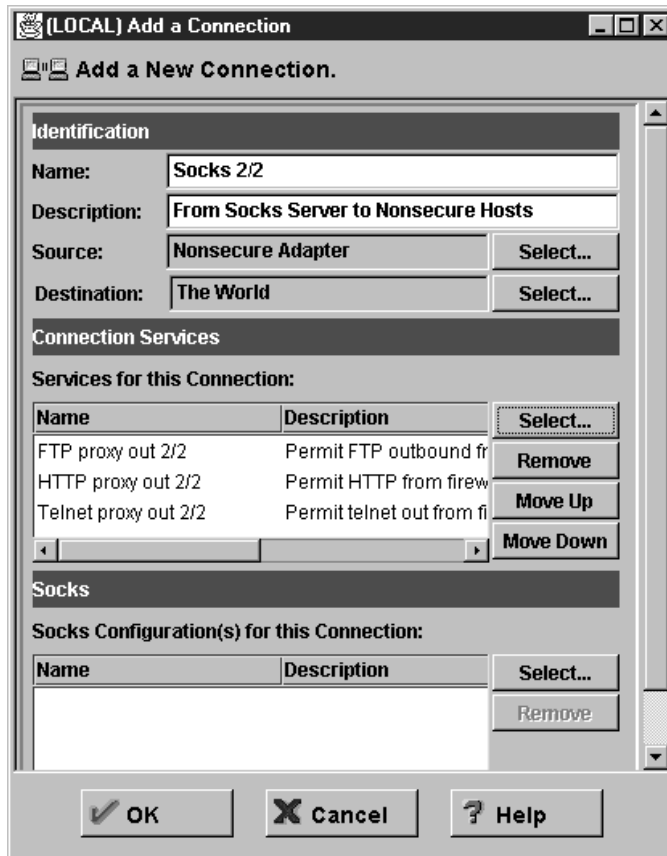


Figure 315. Connection from SOCKS Server to Non-secure Network

In this example, the permitted outbound services are FTP, HTTP and Telnet.

As the last step you specify a SOCKS server configuration for connections between the secure network and the non-secure network. Note that you build a connection with the secure network as the source object, the non-secure network as the destination object, and the SOCKS templates as the services. Figure 316 on page 329 shows the GUI window for such a connection.

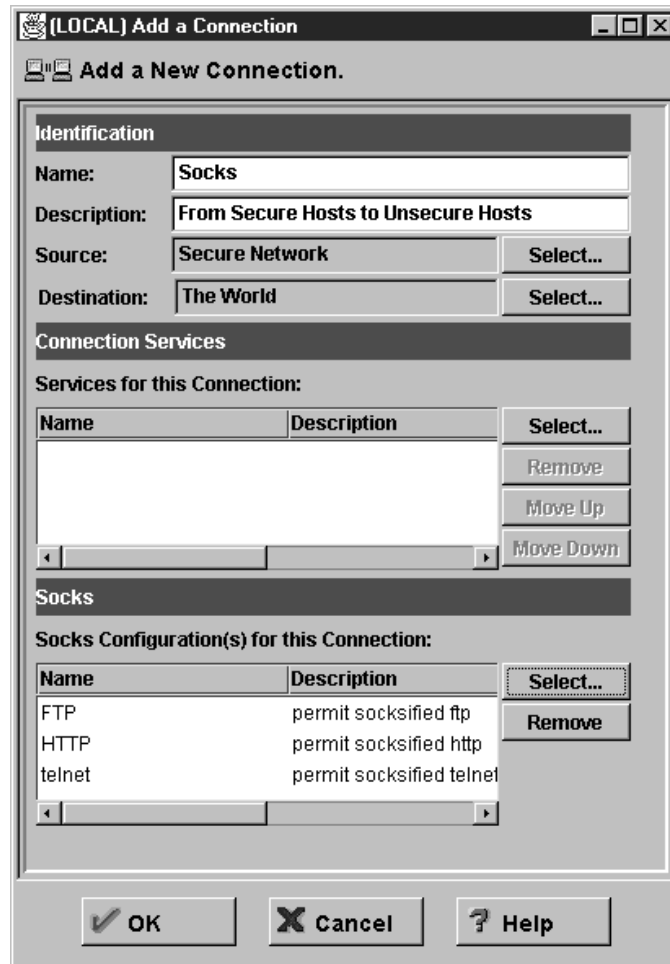


Figure 316. SOCKS Configuration from Secure to Non-secure Network

You click on **Select** to get a list of predefined SOCKS templates. In this example, we selected three templates (FTP, HTTP, and Telnet). The IBM Firewall provides three standard SOCKS templates: HTTP, Telnet and FTP. You can expand the SOCKSified services by creating new templates.

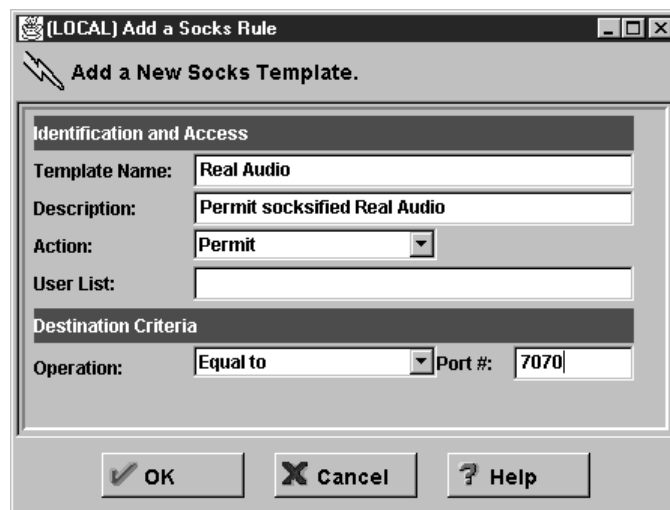


Figure 317. Adding a SOCKS Template

As you can see in Figure 317, the fields to be entered are similar to those for the filter rule definition. The meaning of these fields are as follows:

Template Name is the name of the template.

Description describes the function of the template.

Action is the action to take if a session request matches the conditions in the filter definition. Possible values are permit (allow the session) or deny (refuse session establishment).

User List lists user IDs that this configuration applies to. These are the IDs on the originating host, and they must be listed separated by commas and without blanks.

Operation In the Operation field, enter a logical operator code that represents the logical operation to be performed on the port number. Choose one of Equal to, Not equal to, Less than, Greater than, Less than or equal to or Greater than or equal to.

Port # In the Port # field, enter the number of a port. The port number is used with the Operation field to establish a relationship that must be met. For example, if you enter the Operation Greater Than and Port Number 23, then the port number must be greater than 23 for the rule to be invoked. If this pair is omitted, the line applies to all destination port numbers.

13.7.2.2 Filter Rules

Just for completeness we show you here the filter rules generated from the configuration files for the SOCKS connections. You find a complete discussion in 9.9, "HTTP - World Wide Web Sessions" on page 192).

```
# Description : Permit use of SOCKS from secure network to the firewall
permit 9.24.104.0 255.255.255.0 9.24.104.57 255.255.255.255 tcp gt 1023 eq 1080 secure local inbound
permit 9.24.104.57 255.255.255.255 9.24.104.0 255.255.255.0 tcp/ack eq 1080 gt 1023 secure local outbound

# Description : Permit FTP outbound from firewall to non-secure network
permit 150.53.104.56 255.255.255.255 0 0 tcp gt 1023 eq 21 non-secure local outbound
permit 0 0 150.53.104.56 255.255.255.255 tcp/ack eq 21 gt 1023 non-secure local inbound
permit 0 0 150.53.104.56 255.255.255.255 tcp eq 20 gt 1023 non-secure local inbound
permit 150.53.104.56 255.255.255.255 0 0 tcp/ack gt 1023 eq 20 non-secure local outbound
permit 150.53.104.56 255.255.255.255 0 0 tcp gt 1023 gt 1023 non-secure local outbound
permit 0 0 150.53.104.56 255.255.255.255 tcp/ack gt 1023 gt 1023 non-secure local inbound

# Description : Permit HTTP from firewall to non-secure network
permit 150.53.104.56 255.255.255.255 0 0 tcp gt 1023 eq 80 non-secure local outbound
permit 0 0 150.53.104.56 255.255.255.255 tcp/ack eq 80 gt 1023 non-secure local inbound

# Description : Permit telnet out from firewall to non-secure network
permit 150.53.104.56 255.255.255.255 0 0 tcp gt 1023 eq 23 non-secure local outbound
permit 0 0 150.53.104.56 255.255.255.255 tcp/ack eq 23 gt 1023 non-secure local inbound
```

Figure 318. Filter Rules for FTP, HTTP, and Telnet Using SOCKS

The first pair of rules allows the SOCKS clients to contact the SOCKS server on port 1080. The other rules are protocol-specific and allow the firewall to contact the external servers. These are the same rules used for the proxy server.

13.7.3 Advanced Configuration

The SOCKS configuration file is generated from GUI objects. Depending on your needs, you have three configuration options:

GUI Configuration You can do the most important configuration with the Configuration Client. No additional file editing is needed in this case. You get common defaults for all other configuration options.

Manual Configuration If you want to do all configuration by hand, edit the SOCKS V5 configuration file C:\Program Files\IBM\Firewall\config\socks5.conf. But you may never use the Configuration Client for SOCKS configuration because generating from GUI objects, all your handmade changes will be lost.

Mixed Configuration If you want to use both configuration methods, edit the file C:\Program Files\IBM\Firewall\config\socks5.header.conf. and do all other configuration with the Configuration Client. The socks5.header.conf file is used as a header during generating the socks5.conf file from the GUI objects.

The SOCKS V5 configuration file of the IBM Firewall contains five sections:

- Variables
- Modules
- Routing
- Authentication
- Proxies
- Access Control

The next sections provides an explanation of the major settings. Please refer to the *IBM eNetwork Firewall for Windows NT Reference Version 3 Release 3* and <http://www.aventail.com/> for a full SOCKS V5 configuration.

13.7.3.1 Variables

The following table shows the default values which are applied to the SOCKS server.

Variable	Description	Default Value
SOCKS5_NOVERSEMAP	Don't look up names to go with addresses (works faster this way).	1
SOCKS5_NOSERVICENAME	Don't look up service names to go with port numbers (works faster).	1
SOCKS5_NOIDENT	Disable identd requests.	1
SOCKS5_DEMAND_IDENT	Ensures that a user name is always associated with connection request..	0
SOCKS5_USECLIENTSPORT	Use client port (necessary for proxying streaming-UDP).	1
SOCKS5_BINDPORT	Change the inbound TCP port.	1080

<i>Table 26 (Page 2 of 2). SOCKS V5 Default Behavioral Values</i>		
Variable	Description	Default Value
SOCKS5_RECVFROMANYONE	Allow unsolicited UDP messages.	1
SOCKS5_MAXCHILD	Set maximum number of concurrent children.	64

The format for setting variables is shown in Figure 319:

```
set variable value
```

Figure 319. Variable Definition Format

Where `variable` is a variable from Table 26 on page 331 and `value` is a defined value.

The timeout variable has another syntax, which is different from the SOCKS V5 standard.

```
timeout 15 minutes
```

Figure 320. Timeout Definition Format

13.7.3.2 Modules

This section defines the modules used for authentication:

- `server_password_IBM` (`ibmpwd`)
- `server_cram_IBM` (`ibmcram`)

The first one is used for user ID/password authentication. The second one is used for strong authentication, it supports the Challenge Response Authentication Method (CRAM). The authentication method `null` is defined by default. You don't need additional filter rules for the authentication methods.

If you want to use your own authentication methods, you have to register your modules in this section. The format is shown in Figure 321.

```
module stub filename options
```

Figure 321. Module Definition Format

Where `stub` is a module dependent name prefix for accessing function names, `filename` is the path and name to the module, and `options` are the options for the module. The options can be omitted.

Figure 322 shows the predefined authentication modules from the IBM Firewall.

```
module server_password_IBM C:\PROGRA~1\IBM\Firewall\bin\ibm_gwauthp.mod
module server_cram_IBM C:\PROGRA~1\IBM\Firewall\bin\ibm_gwauthc.mod
```

Figure 322. Module Definition Example

13.7.3.3 Routing

The routing information is used to assign the network interface of multi-homed hosts with a network number, a mask and, if needed, a range of ports. The format is shown in Figure 323.

```
route dest-address dest-port interface-address
```

Figure 323. Interface Definition Format

Where dest-address is an IP address and mask combination to specify a network, dest-port is either a single port or a range of ports, and interface-address is the IP address of a network interface.

Figure 324 shows an example from our scenario.

```
route 150.53.104.56/255.255.255.0 - 150.53.104.56
route 9.24.104.57/255.255.255.0 - 9.24.104.57
```

Figure 324. Routing Information Example

The dash (-) in the two lines indicates that there are no destination port restrictions defined.

13.7.3.4 Authentication

The contents of this section depend on the authentication profile set with the explode.cfg configuration file. If you use the permissive profile, no authentication methods are applied but connections from the non-secure network are denied. This results in ban entries for all non-secure interfaces.

If you use the intermediate profile, SOCKS V5 clients are authenticated while SOCKS V4 clients can establish outbound connections without authentication. This results in an additional null authentication method for connections coming from the secure network.

And if you use the strict profile, only SOCKS V5 clients can establish connections.

The format is shown in Figure 325.

```
auth source-address source-port auth-methods
ban source-address source-port auth-methods
```

Figure 325. Authentication Definition Format

Where source-address is an IP address and mask combination to specify a network, source-port is either a single port or a range of ports, and auth-methods is a list of the defined authentication methods.

Figure 326 shows an example of the intermediate profile.

```
auth 150.53.104.56/255.255.255.0 - ibmpwd,ibmcram
auth 9.24.104.57/255.255.255.0 - null,ibmpwd,ibmcram
```

Figure 326. Authentication Definition Example

The list of authentication methods is applied backwards to clients that are requesting connections. This means the SOCKS server tries the Challenge

Response Authentication Method first and then the username password authentication method. If both methods fail, the connection request is rejected for the 150.53.104.56 interface. 9.24.104.57 is our secure interface. Therefore, SOCKS V4 clients are allowed to use SOCKS without authentication. This is indicated with the null authentication method for this interface.

The dashes (-) indicate that there are no restrictions for the parameter.

13.7.3.5 Proxies

This section contains the proxy rules that tells the server how to connect to the destination. If you don't specify a proxy line, the destination is connected directly. The format is shown in Figure 327.

```
proxy-type dest-host dest-port proxy-addr proxy-port
```

Figure 327. Proxy Definition Format

Where proxy-type is the type of proxy server used (socks4, socks5, or no proxy), dest-host is a combination of a network number and a mask, dest-port is either a single port or a range of ports, proxy-addr is the IP address of the proxy, and proxy-port is the port used for the proxy.

Figure 328 shows an example of a chained SOCKS server.

```
no proxy 9.24.104.0/255.255.255.0 - - -  
socks5 0/0 - 9.25.3.3 1080
```

Figure 328. Proxy Definition Example

The dashes (-) indicate that there are no restrictions for the parameter. The first line ensures that no proxy is used for the local network. The second line forwards all other requests to the outer SOCKS server, which is a Version 5 server.

13.7.3.6 Access Control

The last section contains the SOCKS rules. SOCKS rules can either permit or deny traffic. The format is shown in Figure 329.

```
permit auth cmd src-host dest-host src-port dst-port userlist  
deny auth cmd src-host dest-host src-port dst-port userlist
```

Figure 329. Access Control Definition Format

Where auth is a list of authentication methods, cmd is a command pattern, src-host and dst-host are combinations of a network number and a mask, src-port and dst-port are either a single port or a range of ports, and userlist is a list of users.

Figure 330 shows an example of SOCKSified HTTP.

```
permit - - 9.24.104.0/255.255.255.0 - - 80 -  
deny - - - - -
```

Figure 330. Module Definition Example

The dashes (-) indicate that there are no restrictions for the parameter. The first line allows all users from the network 9.24.104.0 to request connections from any port to port 80 on any host with any defined authentication method.

13.7.4 Example SOCKS Connections

We present you two common connections examples for the SOCKS server. The first one is a direct connection from your internal clients to the firewall SOCKS server. The second example explains proxy and SOCKS chaining. Both scenarios have advantages and disadvantages. They are discussed in the following sections.

13.7.4.1 Browser Connection

In order to use the SOCKS server, your users have to change their SOCKS pointer in your browser to point to the IBM Firewall and the proper port.

To open the Preferences window from the Netscape Navigator Version 4, select **Edit** from the menu and then **Preferences**. From the Category navigation tree select **Advanced** and then **Proxies** (see Figure 309 on page 322). Now switch the radio button to **Manual proxy configuration** and click on the **View** button right from this choice. You should now see the Manual Proxy Configuration window. Figure 331 shows a typical SOCKS configuration.

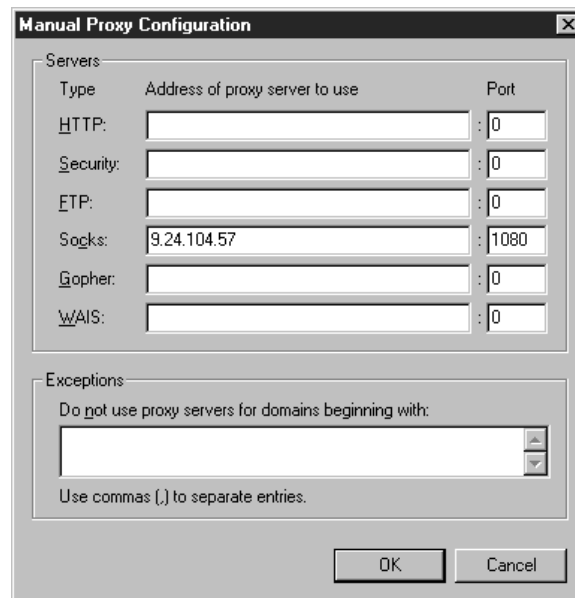


Figure 331. SOCKS Configuration for Netscape Navigator

As you can see, the IBM Firewall is used as the SOCKS server. SOCKS is protocol-independent, therefore you don't have to configure SOCKS in the browser for specific protocols such as HTTP or FTP.

Again, you have no caching of Web pages and FTP downloads.

13.7.4.2 Chaining Proxy and SOCKS

This example describes a chaining of a caching proxy server and the firewall SOCKS server (see Figure 332).

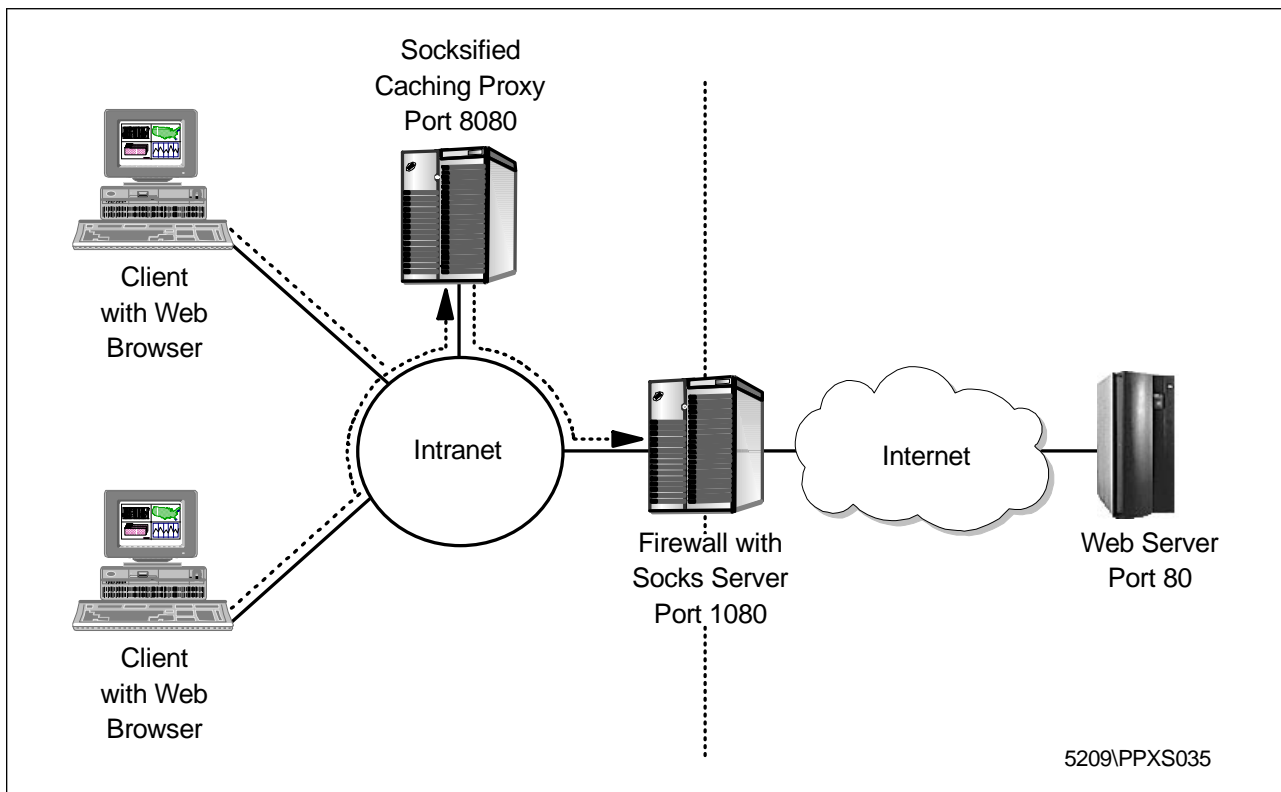


Figure 332. Proxy/SOCKS Chaining Example

We recommend that you use this way for Internet access. We use an internal Web proxy for the chaining proxy server, for example Web Traffic Express from IBM (see 21.3, "IBM Web Traffic Express" on page 529 for further configuration), to cache Web pages and FTP downloads. All internal clients point to this proxy. This proxy server must be SOCKSified to point to the firewall SOCKS server.

13.7.4.3 Chaining SOCKS Servers

You also can chain SOCKS servers, which means to SOCKSify a SOCKS server. See 13.7.3.5, "Proxies" on page 334 and the *IBM eNetwork Firewall for Windows NT User's Guide Version 3 Release 3* for a detailed description.

13.7.5 SOCKS Client Services

In order to make use of a SOCKS server, you need to have a modified *SOCKSified* client program that will direct the session to the SOCKS port on the server and handle the connect request/response sequence or you must *SOCKSify* the whole IP stack.

For general information about SOCKS, consult the SOCKS Home Page at <http://www.socks.nec.com/>.

13.7.5.1 SOCKSified Client Programs

In general, Web browsers (such as the Netscape Navigator or the Microsoft Internet Explorer) provide built-in SOCKS support. SOCKSified versions of many other application clients are available from various Internet sites, for example:

Application	Information Source
Netscape Communicator	http://home.netscape.com/
Microsoft Internet Explorer	http://www.microsoft.com/ie/
Notes 4.5 and Domino	Notes Documentation Database: Working with Lotus Notes and the Internet
IBM Web Traffic Express	http://www.ics.raleigh.ibm.com/WebTrafficExpress/

13.7.5.2 SOCKSified IP Stacks

Several manufacturers of TCP/IP implementations are incorporating SOCKS support into their products (see 21.2, "Aventail AutoSOCKS" on page 521 for more details).

SOCKSified IP Stack are available from the following Internet sites:

Operating System	Applications	Product	Information Source
OS/2 Warp	Any TCP/IP Application		Retrieve Software Updates from the Internet Connection Folder
OS/2 Warp		socks5	http://www.socks.nec.com/socks5.html
Windows	Any WinSock Application	Aventail AutoSOCKS	http://www.aventail.com/
Windows	Any WinSock Application	Hummingbird SOCKS Client	http://www.hummingbird.com/
Windows	Any WinSock Application	NEC SocksCap	http://www.socks.nec.com/sockscap.html
UNIX	Shared Libraries	NEC runsocks	http://www.socks.nec.com/how2socksify.html#runsocks

Chapter 14. Network Address Translation

In order to be assured of any-to-any communication across the Internet, all IP addresses have to be officially assigned by the Internet Assigned Numbers Authority (IANA). This is becoming increasingly difficult to achieve because the number of available address ranges is now severely limited. Many organizations use locally assigned IP addresses, basically used from the three blocks as described in RFC 1918 to avoid colliding with officially assigned IP addresses. These addresses, namely 10/8, 172.16/12 and 192.168/16, will not be routed on the Internet, but you can do it with NAT.

Note: Let's clarify the convention for subnets used above: 192.168/16 is the same as 192.168.0.0 mask 255.255.0.0. The number 16 refers to the number of bits ON from the leftmost bit of the mask.

NAT takes the source IP address of an outgoing packet and translates it to an official address. For incoming packets it translates the official destination address back to an internal address. The basic idea of NAT is to transparently translate a network's internal IP addresses to an official IP address so that they can be routed on the Internet or to hide them for privacy reasons. We now can use NAT as a routing solution for networks that have private address ranges or illegal addresses and want to communicate with hosts on the Internet.

In fact, by implementing the firewall we have already circumvented part of the problem. Clients that communicate with the Internet by using a proxy or SOCKS server do not expose their addresses to the Internet, so their addresses do not have to be translated. However, when we do not want, for whatever reason, to use a proxy or SOCKS server or when proxy and SOCKS are not possible or just explicitly need routing, we can use NAT. For example, proxy and SOCKS V5 servers implemented on IBM eNetwork Firewall for Windows NT cannot be used for all UDP connections.

The NAT solution implemented in the IBM eNetwork Firewall for Windows NT makes sense if you have many hosts communicating from the inside to the outside, but not vice versa. If you want to let users reach a server in your intranet from the Internet, you will have to map the IP address of every inside server to a unique official IP address. If you have many servers, you should consider using a DMZ instead of this approach. If you need to have a one-to-one translation of a network you should use a NAT capable router, like IBM 2210, 2212 or 2216 with V3.1 support.

You should also remember that using NAT means you have to allow routed traffic. Whenever possible, you should use proxy and SOCKS instead of routed traffic for higher security.

14.1 Translation Mechanism

The original so-called traditional NAT as described in RFC 1631 has been implemented in the AIX version of the IBM Firewall. The NT implementation is called NAPT: Network Address Port Translation. This type of NAT is also called a many-to-one connection because you need to have just one officially assigned IP address to communicate to the Internet. For more information have a look at:

<http://www.ietf.org/html.charters/nat-charter.html>

Traditional NAT translates the source IP address of outgoing packets to an official IP address taken out of a pool of official IP addresses. It remembers which official IP address was allocated to the internal IP address. This association can be thought as a NAT session. So for incoming packets it is able to retranslate the official destination IP address to the internal IP address. The drawback to this method is that the number of concurrent active NAT sessions is limited to the number of registered IP addresses available in the pool. Once this limit has been reached, no more NAT sessions can be established until some of the existing sessions have timed-out due to lack of activity.

Figure 333 shows NAT as used under NT.

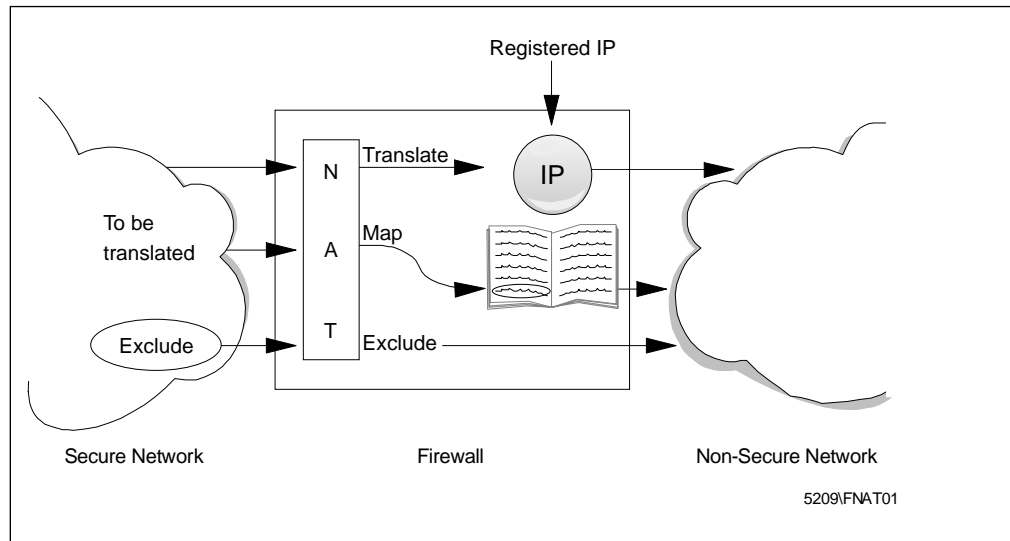


Figure 333. Overview of a NAT Configuration

So, how does NAPT work? For each outgoing packet the source IP address is checked by the NAT configuration rules. If the source IP address matches a translation rule, the address and port translation is performed. NAPT not only translates the internal IP address to an official IP address, but also translates the source port into a free port selected by NAPT. The firewall builds up a table with these two pieces of information to remember the outgoing requests like this:

```
<src IP:src port> -> <new src IP:new src port>  
<9.24.104.242:1378> -> <150.53.104.55:2484>
```

The destination IP address and port remain the same.

When a packet from the Internet arrives at the firewall, NAT looks up the table if it has a matching record for this packet and retranslates it to the original sender's IP address and port.

```
<dst IP:dst port> -> <new src IP:new src port>  
<150.53.104.55:2484> -> <9.24.104.242:1378>
```

For incoming packets, the source IP address and port remain the same.

This translation includes adjusting of the packet's checksum(s).

There are 65536 possible ports; 1024 are reserved for well-known protocols. This means that we still have more than 64000 ports left for communicating to the Internet with just one registered IP address.

This translation works transparently on most TCP and UDP connections. But for certain FTP packets the task is even more difficult because the packets can contain IP addresses in the packet's payload. For example, the FTP PORT command contains an IP address in ASCII. These addresses are also translated correctly by the firewall NAPT including checksum updates and even TCP sequence and acknowledge updates. FTP is the only protocol (that needs such a correction) that is handled by the IBM eNetwork Firewall for Windows NT NAPT. For other protocols that have IP addresses in the packet's payload, you should use traditional proxies or SOCKS if possible. The ICMP protocol is not supported by the current implementation of NAT in the IBM eNetwork Firewall for Windows NT. For example, pinging to nonsecure IP addresses does not work, because ping uses the ICMP protocol.

Basically you need to create filter rules that will normally route packets from a secure network to the Internet and back. NAT will take care of the address translation of the secure addresses. Figure 334 shows how packets flow through the firewall. You will notice that NAT translation will occur for the outgoing packet after the packet has gone through both packet filters (secure and non-secure adapters). This means that all packet filtering is performed on secure side information and you should not use NAT addresses in the filter rules.

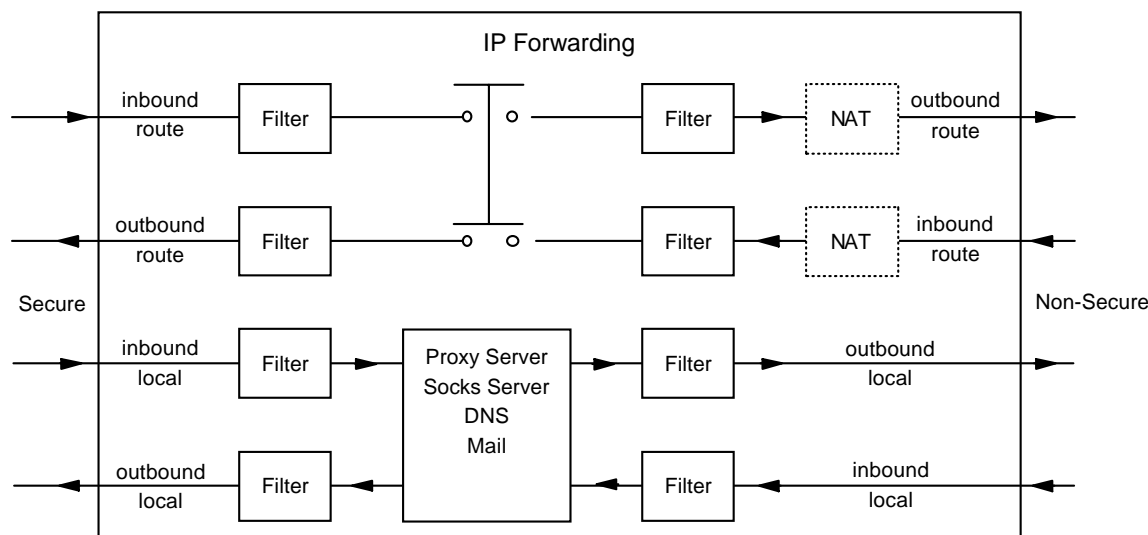


Figure 334. The Flow of the Packets inside the Firewall

14.2 Configuration of NAT

The address translation is done according to NAT rules. The NAT keywords are shown in Figure 335 on page 342.

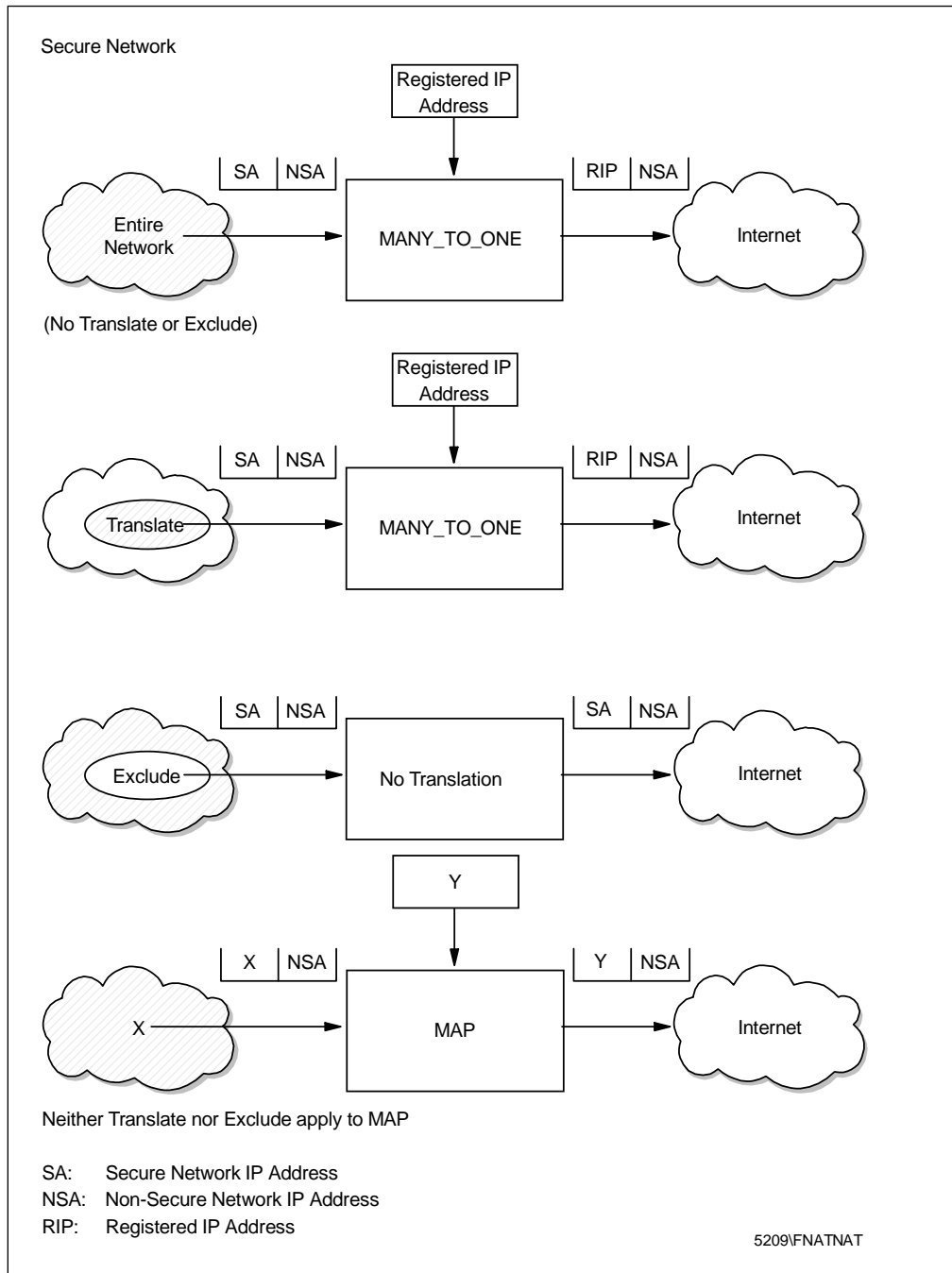


Figure 335. Network Address Translation Keywords

1. Unless we use the TRANSLATE or EXCLUDE commands, MANY-TO-ONE will translate the source IP address of all the outgoing routed packets.

First of all NAT has to know which IP address it may use for the translation. The entry consists of a network address and a timeout value. The timeout value is the number of minutes before NAT deletes an idle address/port pair out of his table. The default value is 15 minutes; the minimum value allowed is 5 minutes. An example is given in Figure 336 on page 343.

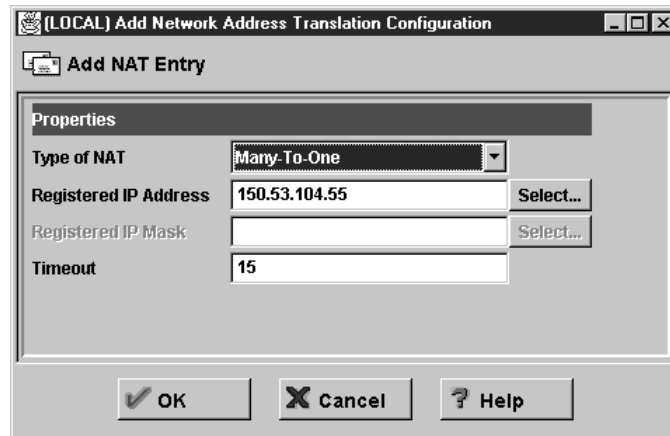


Figure 336. Many-to-one

The screen capture of above registers the IP address 150.53.104.55 to be used for NAT.

2. Define addresses to be translated.

By default all addresses in the secure network are translated by NAT. To limit this, you may specify one or more ranges of addresses which must be translated.

For example, if you want the class C network 9.24.104.0 to be translated, you define it like in Figure 337.

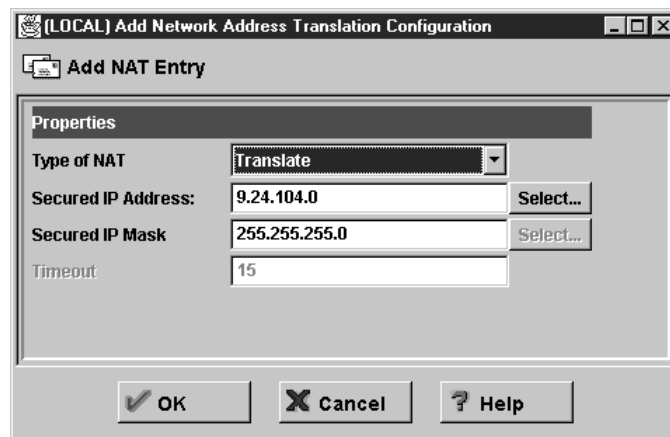


Figure 337. Translate

Once the addresses to be translated by NAT have been set as shown in Figure 337, the secure addresses not in this set will not be translated by NAT.

3. Define addresses to be excluded from translation.

Use this step if you want to exclude some addresses from the range of addresses you specified for translation.

For example, if you don't want to translate the host 9.24.104.70, you define it like in Figure 338 on page 344.

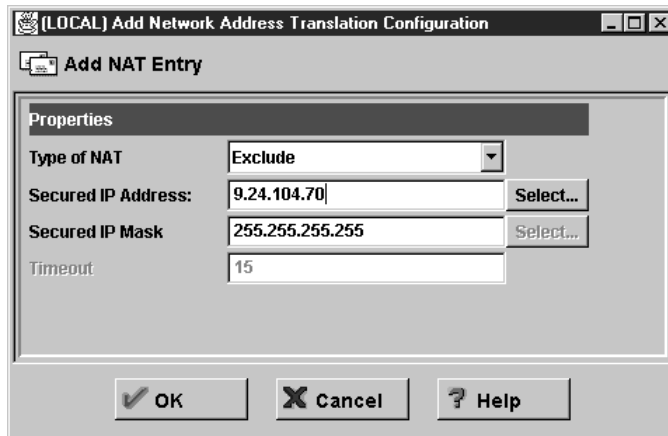


Figure 338. Exclude

4. Define address mappings.

An address mapping allows you to map a secure address to a specific official (registered) IP address. One reason for using MAP instead of MANY-TO-ONE is when you want to allow users from the Internet to access hosts in the secure network without using a proxy. In this case MANY-TO-ONE cannot be used.

For example, if you want the secure address 9.24.104.77 to be translated into 150.53.104.54 then define this mapping like in Figure 339.

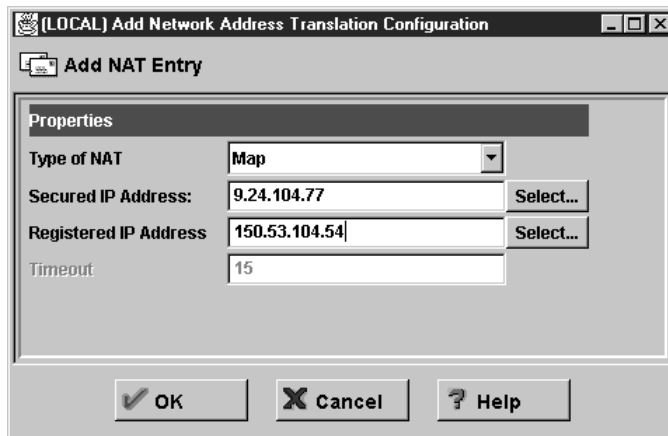


Figure 339. Map

A packet arriving to the nonsecure side of the firewall with destination IP address 150.53.104.54 will be translated into IP address 9.24.104.77. Of course, a similar translation will be done with a packet arriving to the secure side of the firewall with source IP address 9.24.104.77.

Notice that 9.24.104.77 is part of the dynamic translation set defined in Figure 337 on page 343. This is allowed and the static mapping will take precedence over the dynamic MANY-TO-ONE translation that otherwise would have occurred.

5. Activate or update the NAT configuration and logging.

After the initial configuration and after every change you have to activate/update the NAT configuration. You may also decide whether or not to

activate the NAT logging facility. Have a look at Figure 340 on page 345 for the possible options of the Control Activation Panel for NAT.

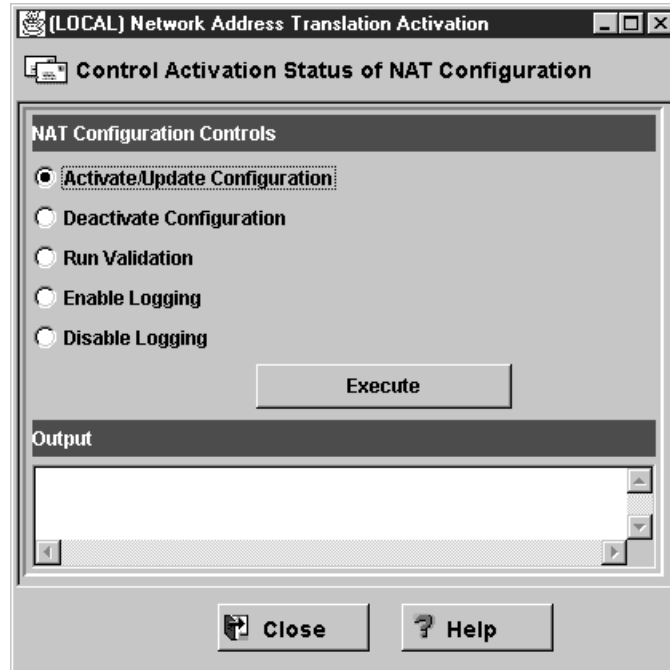


Figure 340. Example Configuration

The definitions stated earlier would appear in the Network Address Translation List as shown in Figure 341.

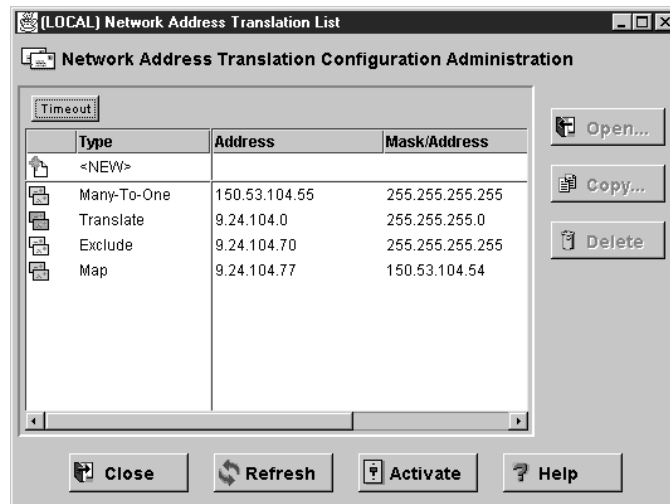


Figure 341. Example Configuration

The use of NAT is transparent to the filter rules. Remember: in the filter rules you have to use your untranslated secure addresses. The NAT address must *not* be used in the filter rules.

For an overview of the possible combinations of the NAT keywords and behavior, see following table:

<i>Table 29 (Page 1 of 2). Possible NAT Rules Combinations</i>	
Active NAT Entry	What NAT Code Does
None	When there are no active NAT configuration file entries, NAT is not active and no secure addresses are translated.
MANY-TO-ONE only	All secure source addresses are translated in all outbound packets.
TRANSLATE only	Outbound packets with secure source addresses matching the TRANSLATE entry are discarded because there is no MANY-TO-ONE entry specifying available external addresses. Outbound packets that don't match the TRANSLATE are allowed through without translation.
EXCLUDE only	Outbound packets with secure source addresses matching the EXCLUDE entry are allowed through without translation. Outbound packets that do not match the EXCLUDE are discarded.
MAP only	Packets with secure source or destination addresses matching the MAP entry are translated. Outbound packets that do not match the MAP are allowed through without translation.
MANY-TO-ONE and TRANSLATE	Outbound packets with secure source addresses matching the TRANSLATE entry are translated because there is a MANY-TO-ONE entry specifying available external addresses. Outbound packets that do not match the TRANSLATE are allowed through without translation.
MANY-TO-ONE and EXCLUDE	Outbound packets with secure source addresses matching the EXCLUDE entry are allowed through without translation. Outbound packets that do not match the EXCLUDE are translated.
MANY-TO-ONE and MAP	Packets with secure source or destination addresses matching the MAP entry are translated. Outbound packets that do not match the MAP entry are translated.
EXCLUDE and TRANSLATE	Outbound packets with secure source addresses matching the EXCLUDE entry are allowed through without translation. Outbound packets with secure source addresses matching the TRANSLATE entry are discarded because there is no MANY-TO-ONE entry specifying available external addresses. Outbound packets that do not match either entry are allowed through without translation even though they do not match the EXCLUDE entry.
EXCLUDE and MAP	Outbound packets with secure source addresses matching the EXCLUDE entry are allowed through without translation. Packets with secure source or destination addresses matching the MAP entry are translated. Outbound packets that do not match either entry are allowed through without translation.

Table 29 (Page 2 of 2). Possible NAT Rules Combinations

Active NAT Entry	What NAT Code Does
MAP and TRANSLATE	Packets with secure source or destination addresses matching the MAP entry are translated. Outbound packets with secure source addresses matching the TRANSLATE entry are discarded because there is no MANY-TO-ONE entry specifying available external addresses. Outbound packets that do not match either entry are allowed through without translation.
MANY-TO-ONE, TRANSLATE, and EXCLUDE	Outbound packets with secure source addresses matching the TRANSLATE entry are translated because there is a MANY-TO-ONE entry specifying available external addresses. Outbound packets with secure source addresses matching the EXCLUDE are allowed through without translation. Outbound packets that do not match either entry are allowed through without translation even though they do not match the EXCLUDE entry.
MANY-TO-ONE, TRANSLATE, and MAP	Outbound packets with secure source addresses matching the TRANSLATE entry are translated because there is a MANY-TO-ONE entry specifying available external addresses. Packets with secure source or destination addresses matching the MAP are translated. Outbound packets that do not match either entry are allowed through without translation.
EXCLUDE, TRANSLATE, and MAP	Outbound packets with secure source addresses matching the EXCLUDE entry are allowed through without translation. Outbound packets with secure source addresses matching the TRANSLATE are discarded because there is no MANY-TO-ONE entry specifying available external addresses. Packets with secure source or destination addresses matching the MAP are translated. Outbound packets that do not match any entry are allowed through without translation.
MANY-TO-ONE, EXCLUDE, and MAP	Outbound packets with secure source addresses matching the EXCLUDE entry are allowed through without translation. Packets with secure source or destination addresses matching the MAP are translated. Outbound packets that do not match either entry are translated.
MANY-TO-ONE, TRANSLATE, EXCLUDE, and MAP	Outbound packets with secure source addresses matching the TRANSLATE entry are translated because there is a MANY-TO-ONE entry specifying available external addresses. Outbound packets with secure source addresses matching the EXCLUDE are allowed through without translation. Packets with secure source or destination addresses matching the MAP are translated. Outbound packets that do not match any entry are allowed through without translation.

14.3 How to Configure Routing when Using NAT

When we define one or more IP registered address(es), we need to associate these addresses with the nonsecure interface of the firewall. The return of the NAT packets from the Internet to the firewall must be enabled via static routing or IP aliasing.

To decide when to use static routing or IP aliasing, we have to distinguish between two cases:

- the NAT registered IP address is in the same subnet as the nonsecure interface.

The nonsecure interface will have to answer with its MAC address to an ARP request for any of the registered IP addresses. We achieve this by defining an IP alias.

- the NAT registered IP address is in a separate subnet of the nonsecure interface

The return of the NAT packets from the Internet to the firewall must be enabled via static routing.

14.3.1 The Registered NAT IP Address is in a Separate Subnet

If your NAT IP registered addresses are in a separate subnet from your nonsecure adapter IP address, you must add a static route that gives the firewall non-secure adapter address as gateway for the NAT addresses as destination. The non-secure router will advertize the route back through the Internet to the backbone routers.

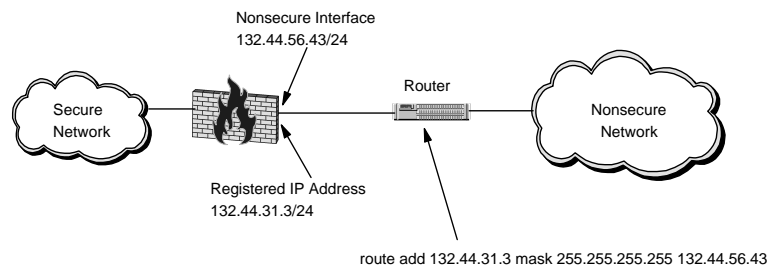


Figure 342. The Registered IP Address is in a Separate Subnet

14.3.2 The Registered NAT IP Address is in the Same Subnet

If your NAT IP addresses are in the same subnet as your nonsecure adapter IP address, you need to get your firewall host to respond to ARP requests for the NAT IP addresses with the MAC address of the nonsecure adapter. On Windows NT you do this by specifying the NAT IP addresses as 'IP aliases' of your nonsecure adapter.

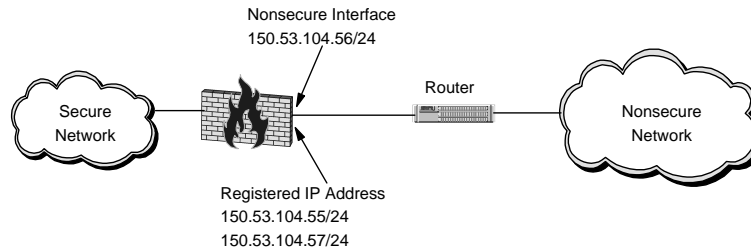


Figure 343. The Registered IP Address is in the Same Subnet

The following procedure will let you define IP address aliases for the configuration of Figure 343.

Open the networks control panel, choose the protocol tab and open the TCP/IP Protocol properties. Choose the **Advanced** button and select the non-secure adapter. Now enter any additional IP addresses you may need for NAT. You have to reboot to activate these IP aliases. An example is shown in Figure 344.

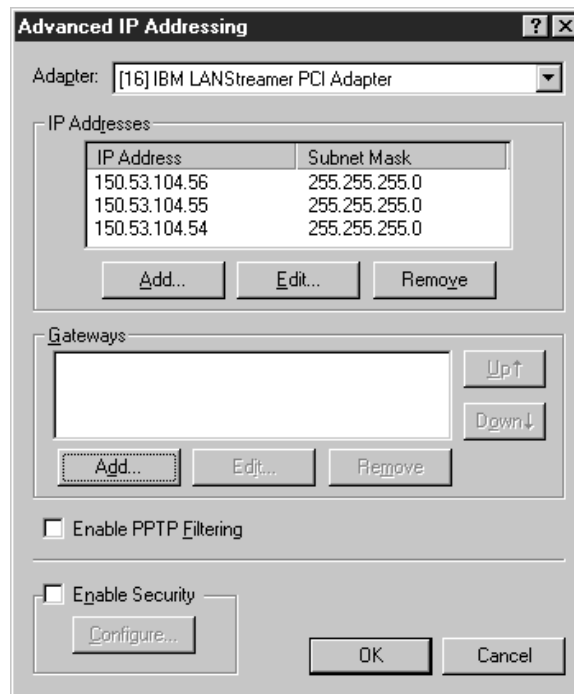


Figure 344. Defining IP Aliases

14.3.3 Maximum Transmission Unit

The Maximum Transmission Unit (MTU) for a host is the maximum datagram size it can handle. When one host sends data to another host it is preferable that the datagrams have the largest size that does not require fragmentation anywhere along the path from the source to the destination. This datagram size is referred to as the Path MTU (PMTU).

At the startup of a TCP connection only the MTU values between endpoints are usually considered. So, it is possible that a packet arrives at an intermediate host that has a smaller MTU. This is solved by fragmenting or sending ICMP type 3 code 4 packets when fragmentation is not allowed.

However, NAT does not support ICMP so when using NAT the firewall must have the minimum MTU on the path. Also, all hosts and routers directly connected to the non-secure interface must have this MTU.

The default MTU for the IBM eNetwork Firewall for Windows NT is based on the network topology. Below is a list of maximum MTU sizes for different media.

Network	MTU(Bytes)
16 Mbit/Sec Token Ring	17914
4 Mbits/Sec Token Ring	4464
FDDI	4352
Ethernet	1500
IEEE 802.3/802.2	1492
X.25	576

You cannot use the above values if you are connected to the Internet. For a connection to the Internet we recommend 1440 bytes, which is IEEE 802.3 minus 52 bytes for the packet headers. This should give you a reasonable throughput without having problems passing these packets to most of the other media that use a larger MTU. This can be done with:

- 1) Run the Registry Editor.
- 2) From the HKEY_LOCAL_MACHINE subtree, go to the following key:
 \SYSTEM\CurrentControlSet\Services\- 3) Select **Edit > Add Value**.
- 4) Add a new DWORD with the name MTU and set it to 1440
- 5) Click **OK**.
- 6) Quit the Registry Editor.
- 7) Shut down and restart Windows NT.

For more information about this topic you may browse through the Microsoft Knowledge Base searching for "Maximum Transfer Unit" under

<http://support.microsoft.com/support/search/>

14.4 Timeout Value

The default timeout value for NAT dynamically translated (many-to-one) connections is 15 minutes. For example, if your NAT Telnet connection is idle more than 15 minutes, your NAT address/port will be released from the table. Effectively it means that your Telnet connection is lost. NAT will log these timeout IP address releases to the syslog; the log message number is ICA9047.

The timeout value does not apply to MAP connections.

You may increase the timeout value if you expect the clients to have longer idle times than 15 minutes. Bear in mind that even when the connection is closed, the address/port pair will be reserved for an amount of time equivalent to the timeout value. If you run out of address/port pairs, you will get error message ICA9046.

14.5 Example Configurations for Using NAT

Now we will present some useful scenarios with NAT.

14.5.1 Using NAT to the Internet

Let's assume we have a setup as in Figure 345.

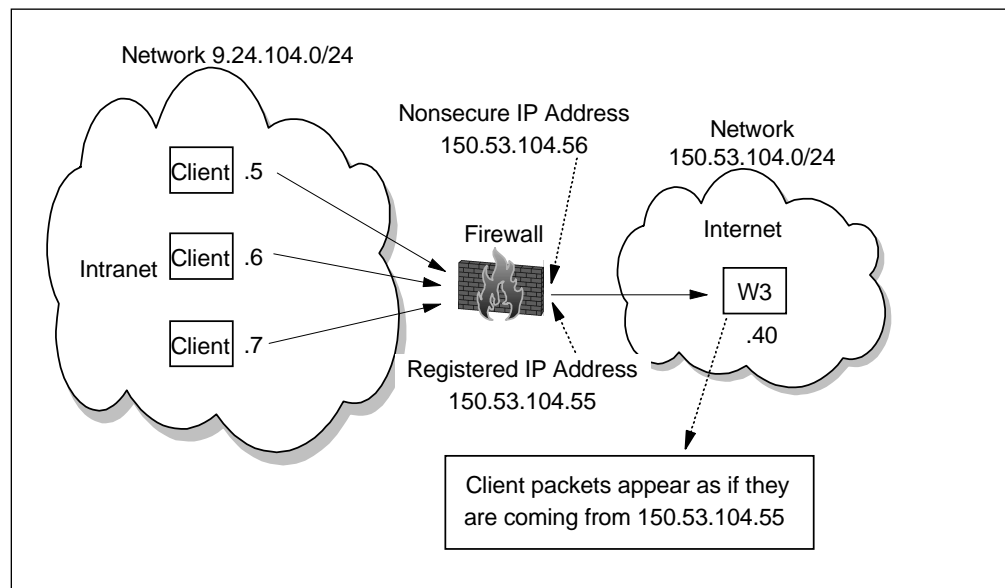


Figure 345. Basic NAT Configuration

For this scenario IP forwarding was activated in the firewall. The client's default gateway was the secure interface of the firewall as we did not have a router in our test configuration; in a real case, you would not use the firewall as a gateway. Instead of using the firewall as a gateway, we could have added a route statement in the secure network clients stating that to reach the 150.53.104.0 network (see Figure 345), the gateway should be the secure interface of the firewall.

```
route add 150.53.104.0 mask 255.255.255.0 'secure_interface_IP_address'
```

There are no proxies or SOCKS configured for the Web browsers on the client. We added the "Many-to-one" registered IP address 150.53.104.55 in the NAT configuration as it can be seen in Figure 336 on page 343. Remember to activate this configuration in the NAT Activation Panel. From now on all secure IP addresses will be translated. Then we defined filter rules that allowed direct outgoing HTTP traffic from the secure network to the world. Pre-defined services for "HTTP direct out" ship with the firewall; just modify and activate.

The filter rules are:

```
# Service : HTTP direct out
# Description : Permit HTTP from secure network directly to non-secure network
permit 9.24.104.0 255.255.255.0 0 0 tcp gt 1023 eq 80 secure route inbound l=n f=y
permit 9.24.104.0 255.255.255.0 0 0 tcp gt 1023 eq 80 non-secure route outbound l=n f=y
permit 0 0 9.24.104.0 255.255.255.0 tcp/ack eq 80 gt 1023 non-secure route inbound l=n f=y
permit 0 0 9.24.104.0 255.255.255.0 tcp/ack eq 80 gt 1023 secure route outbound l=n f=y
```

When we tried to access the non-secure Web server, it worked fine. The access log of the Web server only showed connections from 150.53.104.55, which is exactly what we wanted.

14.5.2 Mapping a Server

Let's assume we have a setup like Figure 346. You want to allow users in the Internet to access your W3 server that resides in your secure network.

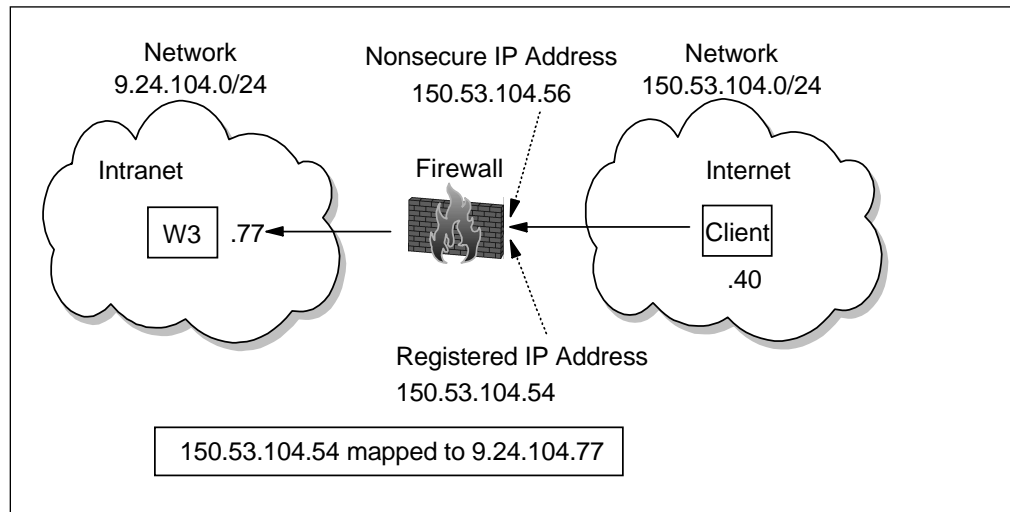


Figure 346. Mapping IP Addresses with NAT

When you don't have a DMZ for whatever reason, NAT can provide some kind of protection, but not much, as opposed to set W3 outside the secure network. NAT will provide some protection but not much: the Internet users can start a session to W3.

Even if you plan to set up a DMZ it might be a good idea to use NAT anyhow, since you protect your servers a little more by hiding their real addresses. However, as you don't gain additional IP addresses with this you might think about a "One-to-many" NAT mapping, where each port can be mapped to a different IP address. This can't be done with the firewall itself but in combination with the Network Dispatcher. You will find an example in 20.3, "ND Advanced Usage: NAT One-to-Many Simulation" on page 513.

Back to our example, we took the "HTTP direct out" templates and configured a "HTTP direct in" where we just exchanged the secure and non-secure interfaces on the four connections. With this we defined a new connection from "The World" to the secure IP address of the Web server. Remember not to specify any NAT IP addresses in the rules.

Checking the filter rules the following appeared:

```
#      Service : HTTP direct in
# Description : Permit HTTP from nonsecure network directly to secure network
permit 9.24.104.77 255.255.255.255 0 0 tcp/ack eq 80 gt 1023 secure route inbound l=n f=y
permit 9.24.104.77 255.255.255.255 0 0 tcp/ack eq 80 gt 1023 non-secure route outbound l=n f=y
permit 0 0 9.24.104.77 255.255.255.255 tcp gt 1023 eq 80 non-secure route inbound l=n f=y
permit 0 0 9.24.104.77 255.255.255.255 tcp gt 1023 eq 80 secure route outbound l=n f=y
```

Now we made the mapping as in Figure 339 on page 344. Before you can use this mapped IP don't forget to set it in the network preferences as shown in Figure 344 on page 349. When you add such IP addresses don't forget to reboot or it won't work. You can see what IP addresses are active when you type **ipconfig** on a command line.

The last step was to activate the new NAT configuration and the new rules. Then we used the client's Web browser, which doesn't have any gateway, proxy or SOCKS configured, to see what happens. Accessing the IP address 150.53.104.54 worked as expected. The server access log showed that 150.53.104.40 accessed the service.

14.5.3 Connecting two Networks with Same IP Addresses

Let's assume we have a setup like in Figure 347.

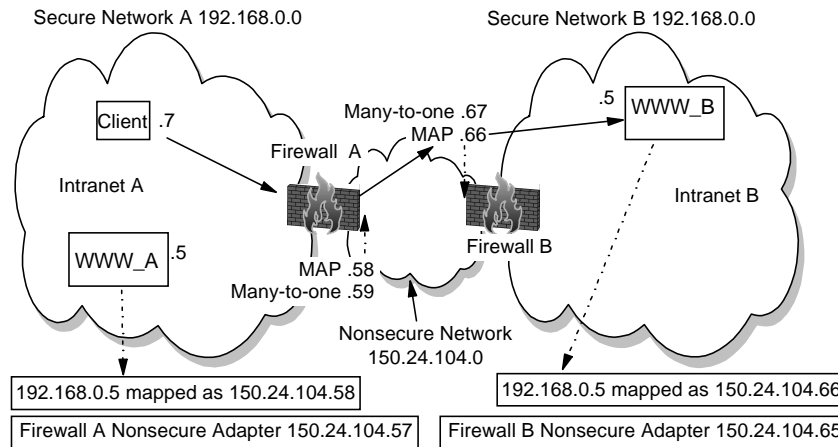


Figure 347. Connecting Networks with the Same IP Addresses

Since most private networks will use the recommended 10/8, 172.16/12 and 192.168/16 network addresses you may run into the situation that you have to connect to networks that have defined the same IP addresses as yours. NAT is a solution to allow the impossible - connecting private networks without changing either network's structure.

Note: This example configuration is not useful if you want to have a 1:1 mapping between the networks, that is everyone on one side can access everybody on the other side. If you need this you should consider using a router capable of doing such a thing.

On firewall A we added a "Many-to-one" rule with the registered IP address 150.24.104.59 and also a "Map" with the registered IP address 150.24.104.58 mapped to the internal IP address 192.168.0.5.

On firewall B we added a "Many-to-one" rule with the registered IP address 150.24.104.67 and also a "Map" with the registered IP address 150.24.104.66 mapped to the internal IP address 192.168.0.5.

For every machine on network B with a duplicate IP address on network A that should be accessible from network A, you need an additional mapped IP address on firewall B. If this process proves impractical, the administrators of these networks could consider using a router instead to handle the 1:1 access required.

Figure 347 has solid arrows that follows the IP traffic from a client in network A to the server WWW_B in network B.

We will follow this and see how the IP addresses are changed along the route. Our client's IP address is 192.168.0.7 and our destination will be 150.24.104.66; this is the mapped WWW_B.

The packets from the client to the firewall A will have these addresses:

```
<src IP:src port> <dst IP:dst port>
<192.168.0.7:1378> <150.24.104.66:80>
```

Port 1378 is an ephemeral port.

When the firewall A does the "Many-to-one" NAT it will be like this:

```
<src IP:src port> <dst IP:dst port> -> <new src IP:new src port> <dst IP:dst port>
<192.168.0.7:1378> <150.24.104.66:80> -> <150.24.104.59:2484> <150.24.104.66:80>
```

Port 2484 was randomly chosen by NAT.

Now it reaches firewall B as:

```
<src IP:src port> <dst IP:dst port>
<150.24.104.59:2484> <150.24.104.66:80>
```

On firewall B the NAT "Map" will take place:

```
<src IP:src port> <dst IP:dst port> -> <src IP:src port> <new dst IP:dst port>
<150.24.104.59:2484> <150.24.104.66:80> -> <150.24.104.59:2484> <192.168.0.5:80>
```

You will see that only the destination IP is changed. No port changes will be done here; the packet should arrive at the WWW server with port 80.

The following table summarizes the changes experienced by the packet as it flows from source to destination.

After leaving	Source IP Address and Port	Destination IP Address and Port
Client A	192.168.0.7:1378	150.24.104.66:80
Firewall A	150.24.104.59:2484	150.24.104.66:80
Firewall B	150.24.104.59:2484	192.168.0.5:80

For the way back this process will simply be reversed:

After leaving	Source IP Address and Port	Destination IP Address and Port
WWW_B	192.168.0.5:80	150.24.104.59:2484
Firewall B	150.24.104.66:80	150.24.104.59:2484
Firewall A	150.24.104.66:80	192.168.0.7:1378

Firewall B should have this rules to allow clients of network A to access WWW_B:

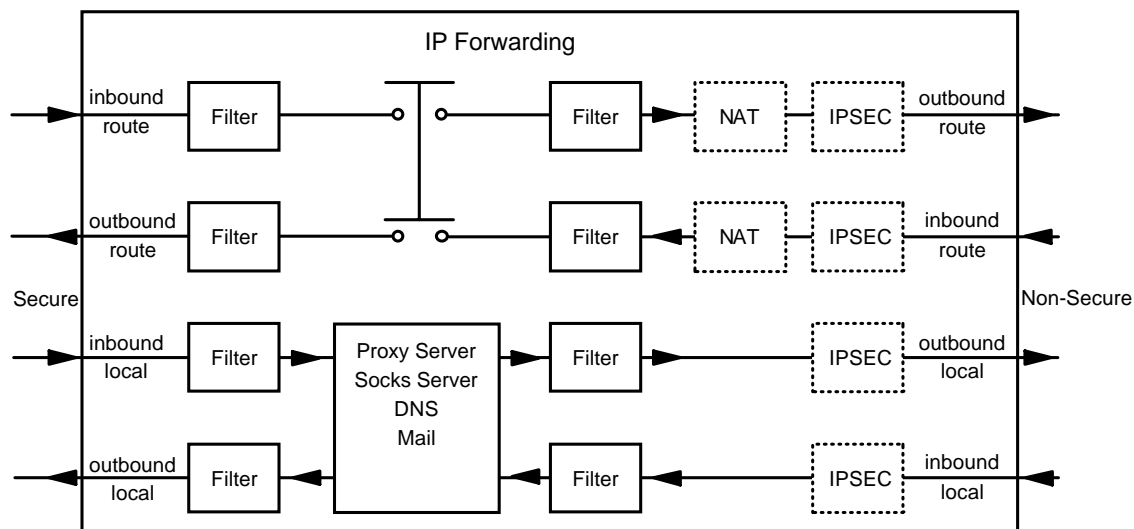
```

# Service : HTTP direct in
# Description : Permit HTTP from network A directly to WWW_B
permit 150.24.104.59 255.255.255.255 192.168.0.5 255.255.255.255 tcp gt 1023 eq 80 non-secure route inbound l=n f=y
permit 150.24.104.59 255.255.255.255 192.168.0.5 255.255.255.255 tcp gt 1023 eq 80 secure route outbound l=n f=y
permit 192.168.0.5 255.255.255.255 150.24.104.59 255.255.255.255 tcp/ack eq 80 gt 1023 secure route inbound l=n f=y
permit 192.168.0.5 255.255.255.255 150.24.104.59 255.255.255.255 tcp/ack eq 80 gt 1023 non-secure route outbound l=n f=y

```

14.6 NAT and Virtual Private Networks (VPNs)

One very basic thing to remember with NAT is that the NAT IP addresses will not be used in your local filter rules and there is no exception when using NAT together with VPN. Figure 348 shows where NAT and VPN take place in the IBM eNetwork Firewall for Windows NT.



5209\FVPN048

Figure 348. Where NAT and IPSEC Functions are Performed

But your NAT IP address will be visible in the Internet and, of course, to the other firewall where the tunnel terminates.

For example we take the scenario in Figure 347 on page 353 and build a tunnel between the firewalls. In the tunnel's filter rules of firewall A the NAT IP address 150.24.104.67 (registered IP address of firewall B) has to be used, and on firewall B it will be 150.24.104.59 (registered IP address of firewall A) that will have to be used.

It will be as easy as this; just remember that NAT registered IP addresses of a given firewall should never appear in its own filter rules.

See 15.5.1.2, "Using NAT with IPSEC" on page 387 for more details.

Chapter 15. Virtual Private Network

In this chapter we discuss the secure IP tunnel. It is a mechanism provided by IBM eNetwork Firewall for Windows NT in order to allow secure communications between secure networks over a nonsecure intervening network like the Internet. It constructs a virtual private network (VPN) between two different sites providing authentication and encryption. A redbook with more detailed information about virtual private networks is A Comprehensive Guide to Virtual Private Networks, Volume I: IBM Firewall, Server and Client Solutions, SG24-5201.

15.1 Secure IP Tunnel Standards - Interoperability

Secure IP tunnel products have existed in the market for years. However, due to the lack of an IP security standard, these products were proprietary in nature (that is, they can only establish secure tunnels with their own product). IBM and other organizations were actively involved in the development of interoperability standards that would allow firewalls from different manufacturers to establish tunnels between them. The basis for these standards is a group of RFCs of the Internet Engineering Task Force (IETF) IP Security Protocol (IPSec) working group. The charter for the IPSec group, plus links to the IPSec RFCs can be found at <http://www.ietf.org/html.charters/ipsec-charter.html>.

Basically, IPSec is a network layer security protocol that will provide authentication, integrity checking and encryption to IP datagrams. It can provide automated tunnel setup, automated secure distribution and automated key refresh using the Internet Key Exchange protocol (IKE). IPSec is the IETF-chosen security framework for both IPv4 and IPv6 environments, and is recommended as the security for PPTP, L2F and L2TP tunnels. IPSec defines two mechanisms to achieve these security objectives. They are the IP Authentication Header (AH) and the IP Encapsulating Security Payload (ESP). The details of AH and ESP can be found in RFC 2402 and RFC 2406 respectively, and the overall architecture of IPSec is described in RFC 2401. The supported data integrity algorithms are Hashed Message Authentication Code (HMAC) using Message Digest 5 (MD5) or Secure Hash Algorithm (SHA); these are described in RFC 2403 and RFC 2404. The supported encryption algorithm Data Encryption Standard (DES) is described in RFC 2405. Triple DES is described in RFC 2451.

IBM eNetwork Firewall for Windows NT provides two kinds of secure IPSEC tunnels to cater to different situations.

We can establish manual tunnels with dynamic or static filters rules, both using the IPSec standard. This has to be done manually by exporting and importing the tunnel configurations. They do not support Key Exchange; this means that the keys are set at tunnel definition and do not change throughout the life of the tunnel.

To establish a IPSEC tunnel between the IBM eNetwork Firewall for Windows NT and another host, the partner node must support the new IPSEC headers. IPSEC tunnels can be established between an IBM eNetwork Firewall for Windows NT and:

- An IBM eNetwork Firewall for Windows NT
- OS/390 Firewall Technologies V2R7 or higher

- eNetwork Firewall for AIX V4.0 (when available); it will not work with the eNetwork Firewall for AIX 3.3
- AIX 4.3 Operating System (supports the old and new IPSEC headers)
- AS/400 Operating System V4R4 (supports new IPSEC headers)
- 2210/2212/2216 Routers
- Other vendor's software that supports the new IPSEC headers

The new IPSEC headers are described in RFCs 2401, 2402, 2403, 2404 and 2405; the old ones are described in RFC 1825, 1826 and 1827.

The IPsec RFCs are very broadly based. In order to bring the technology to fruition more rapidly, RSA Data Security, Inc. convened a group of leading firewall manufacturers in an initiative called S/WAN. The objective of S/WAN is to demonstrate interoperability using a current draft of the IPsec standards. At the time of writing, the S/WAN is executing the second round of their interoperability test. The results of the interoperability tests between the IBM eNetwork Firewall for Windows NT and other firewall vendors are not available yet. More details about the S/WAN initiative can be found under the RSA home page located at <http://www.rsa.com/rsa/SWAN>.

In this second round interoperability test, not only the IPsec, but for the first time, key exchange is tested. The participants are in two groups, each supporting a key management standard ISAKMP/Oakley and SKIP. AH and ESP are designed to be independent of the key management protocol. However, they can be coupled to any key management protocols via the Security Parameters Index (SPI) in their header.

The IP Security Protocol (IPsec) is the IETF-chosen security framework, using the Internet Key Exchange protocol (IKE, previously called ISAKMP/Oakley).

15.2 Operation of the Secure Tunnel

The secure IPSEC tunnel relies on symmetric-key cryptography to enforce data security. This means that the firewalls at each end of the tunnel have a *shared secret* in the form of an encryption key known to both of them. Using this key, the secure IPSEC tunnel provides two different types of security:

1. Authentication, in which the sending firewall appends a message authentication code (MAC) to the messages it sends through the tunnel. The MAC is constructed from the message contents and the encryption key using a one-way hash function. The receiving firewall performs the same operation and, if the MAC matches, it knows that the message is authentic and has not been altered while transported over the nonsecure network.
2. Encryption, in which the data within the message is encrypted using the secure key, so that it cannot be viewed in transit.

Authentication and encryption can be used independently. In fact, you can enable or disable the two features for each tunnel. A typical scenario will have multiple secure networks (for example, branches of a company that are in different cities) with tunnels between them in order to protect the information. There may be more than one tunnel between a single pair of nodes, which might be useful for different encryption and authentication choices.

For example, your computer department may wish to monitor machines in the finance department using SNMP. In this case, the information itself is not sensitive, but you want to be sure that it is accurate, so you could use a tunnel that provides only authentication.

However, you also want the computer department to send mail to the finance department and you would like to protect this mail from being read in the nonsecure network. This would require a second tunnel providing both authentication and encryption.

When a packet has to go from one secure network to another secure network through the IPSEC tunnel, the whole IP packet will be encrypted and authentication data is created at the first end and sent in a new IP packet to the second end of the tunnel. Note that the packet is not sent using the normal IP protocols (TCP or UDP), but using a special security protocol (AH or ESP). In Figure 349 we show a black border around the original IP packet to show that it is being protected in the nonsecure network by the secure IPSEC tunnel.

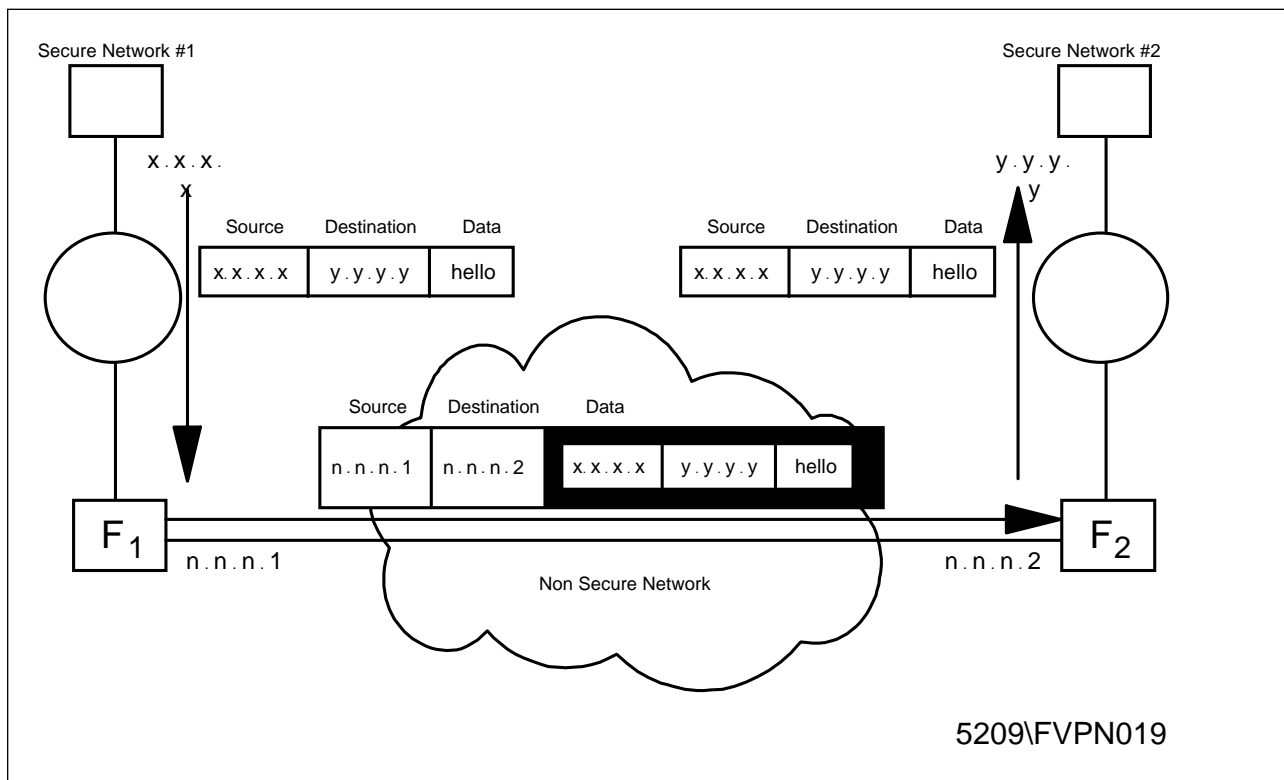


Figure 349. Operation of the Secure IPSEC Tunnel

When a packet leaves a host in secure network #1 the source IP address is x.x.x.x with destination IP address y.y.y.y. In the firewall F1 the packet is encapsulated and the new IP addresses are those of the nonsecure adapters of both firewalls. When the packets reaches the firewall F2, the original packet is restored.

15.3 Implementing the Secure IPSEC Tunnel

In order to configure a tunnel with dynamic or static filters with the IBM eNetwork Firewall for Windows NT you will have to follow these steps:

1. Add the tunnel definition in one node (tunnel owner)
2. Export the tunnel definition to a file
3. Transfer the tunnel definition file to the partner node
4. Import the tunnel definition in the partner node
5. Activate the tunnel at both ends
6. Specify which protocols you want to tunnel using filtering rules and activate the rules. (for tunnels with static filter rules only)
7. Reactivate the tunnel when the life time has expired.

You also have to consider that you need the following prerequisites:

1. IP forwarding enabled in both firewalls for routed traffic
2. Coherent IP addresses in both secure networks (for example, you cannot use the same private IP addresses), If this is not the case, we will have to use Network Address Translation (NAT); see Chapter 14, "Network Address Translation" on page 339 and 15.5.1.2, "Using NAT with IPSEC" on page 387 for details.
3. Proper routes in the clients (they point to the firewall for addresses in the other secure network)
4. Name resolution for the remote networks (this is important if you want to pass hidden DNS information through the tunnel)

We will describe each implementation step in turn.

15.3.1 Adding the Tunnel Definition in One Node

The tunnel has to be created manually; we can create a tunnel with dynamic or a static filter rules. Tunnels with dynamic filters are easy to configure; they have fixed filter rules which are generated automatically when the tunnel is used. The disadvantage is that we have no time of day filtering and no logging. Tunnels with static filter rules have user defined filter rules and are more granular to control.

15.3.1.1 When to Use Each Type of Tunnels

Based on the descriptions of above, you may want to use an IPSEC tunnel with dynamic filter rules in the following cases:

- trusted location; we have to remember that once the tunnel is established, the remote hosts have almost the same access to our secure network resources as any host in our network, only limited to the end points of the tunnel.
- testing purposes, in order to avoid any possible filter rules errors

You may want to use an IPSEC tunnel with static filter rules in the following cases:

- we want to allow only specific protocols to flow through the tunnel
- we have more than one tunnel between two locations and we want to distribute the traffic on each tunnel based on protocols

- we have more than one tunnel between two locations and we want to send encrypted traffic through one tunnel and authenticated traffic through the other
- we want to have the possibility of logging the packets that flow through the tunnel
- there is a network address translation (NAT) at one or both ends of the tunnel
- we want limit the use of the tunnel to certain hours of the day

We will now configure a tunnel with dynamic rules, using the configuration shown in Figure 374 on page 381. A tunnel with static rules will be configured in 15.5, “Virtual Private Network Scenarios” on page 380.

When we select **Virtual Private Network** from the main menu, we get the Administrator Menu (Figure 350), where we can open, copy, delete, import, export, activate and deactivate tunnels.

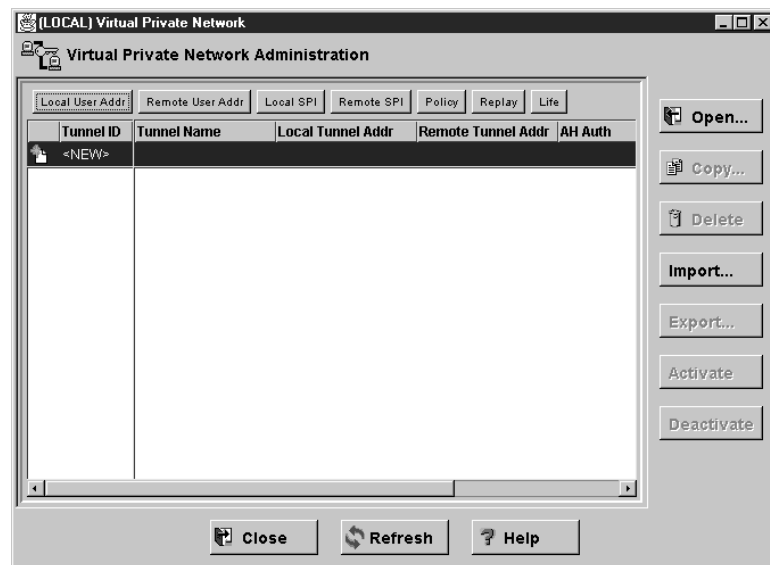


Figure 350. The Virtual Private Network Administration

When we want to create a tunnel for the first time we can only select **Open** or **Import**; **Open** to create a new tunnel, **Import** if we will import the tunnel definitions from another tunnel owner.

When we create the tunnel we have to define, using the GUI, the characteristics of the tunnel: the tunnel type, the addresses of both ends of the tunnel, the authentication/encryption desired, and the parameters for the session key. When creating a tunnel with dynamic rules we also have to define the local and remote addresses. Figure 351 on page 362 shows the GUI screen.

At this point it is very important to double-check the addresses of the tunnel. When we later import them at the partner end, there is no validation against the addresses.

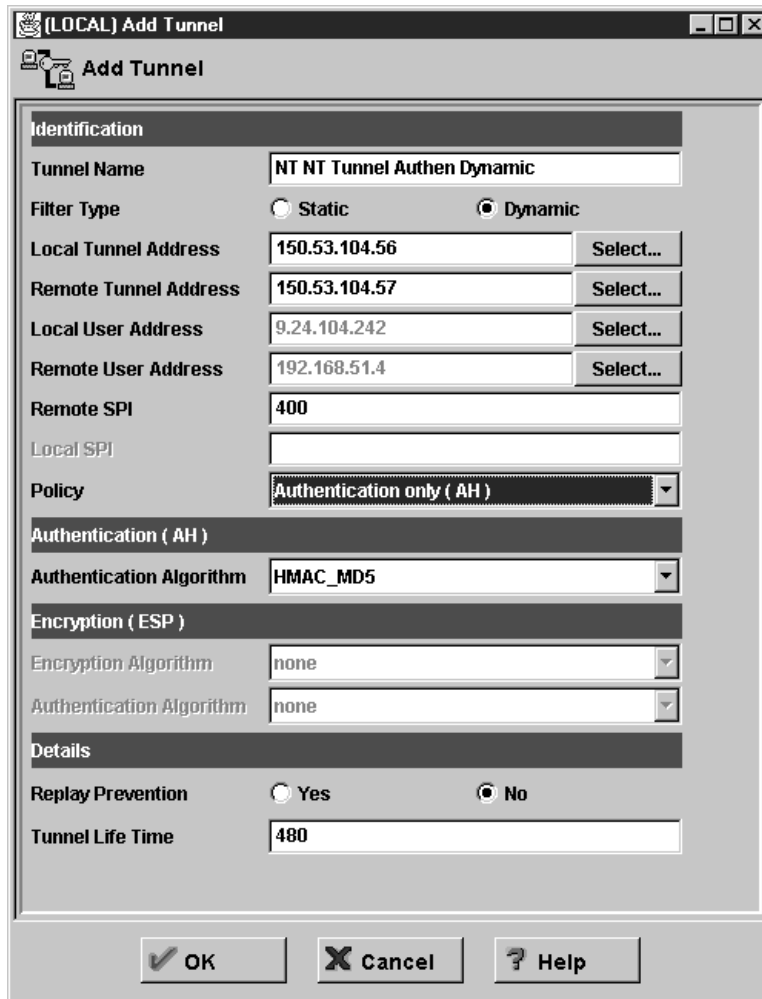


Figure 351. Adding a Tunnel Configuration

In the tunnel's definition, the following fields have special importance:

- Tunnel Name: Enter the name of the tunnel.
- Filter Type: This can be either static or dynamic.

When we select **Dynamic**, the Firewall will generate dynamic filter rules each time the tunnel is activated. This means that all the traffic between the specified users will be accepted in the tunnel.

When we select **Static**, we must create the filter rules for the tunnel. We have the possibility to create multiple tunnels; we can decide which protocol will use each tunnel. These filter rules may be more selective; for example, allow specific protocols traffic through the tunnel.

- Local Tunnel Address: IP address of the nonsecure interface of the local firewall. Clicking **Select** gives us the list with the interfaces.
- Remote Tunnel Address: IP address of the remote partner's nonsecure interface. Clicking **Select** gives us the list of the network objects.
- Local User Address: IP address of the secure network or secure host who will use the tunnel. Clicking **Select** gives us the list with the network objects (only used for Dynamic Filter type).

- Remote User Address: IP address of the remote network or host to which we will connect through the tunnel. Clicking **Select** gives us the list with the network objects (only used for Dynamic Filter type).
 - Remote SPI: Specifies the security parameter index (SPI) value the tunnel partner will use. The value entered must be greater than 255. The definition of SPI is described in RFC 2401. Basically, the SPI in conjunction with the target address will uniquely identify the set of security information (such as encryption key(s), key lifetime, etc.) for your tunnel partner. You should check with the tunnel partner and obtain an unassigned SPI from it.
 - Local SPI: Specifies the security parameter index (SPI) value the tunnel owner will use. The value is entered automatically.
 - Policy: Define which policy we will use; we can select Authentication (AH), Encryption (ESP) or both (ESP/AH).
 - Authentication Algorithm (AH): Enter the type of authentication algorithm we will use; the types available are: HMAC_MD5 AND HMAC_SHA.
 - Encryption Algorithm (ESP): Enter the type of authentication algorithm and encryption algorithm we will use. The encryption types are CDMF, DES_CBC, 3DES_CBC or none, depending on the country version of the firewall. For authentication we can select HMAC-MD5, HMAC-SHA or none.
- Note:** We cannot select **none** for both authentication and encryption with ESP.
- Replay Prevention: Replay prevention uses a sequence counter to prevent old packets from being replayed as a form of attack. The firewall tunnel will always send out replay prevention information in the packet. Check **Yes** to turn on the replay checking of the receiving side.
- Note:** If we have a replay-enabled tunnel defined between an IBM 2210 or 2216 router and a Windows NT Firewall, the tunnel connection will be deactivated if you reboot the firewall. We must reboot the 2210 or 2216 router to activate the tunnel.
- Tunnel Lifetime: specifies the time in minutes that the tunnel will be operational. Put a value in the entry field. The default is 480 (eight hours) and the maximum time allowed is 99999. If you specify 0, it means that the tunnel will not time out; the lifetime is unlimited.

Note: After a reboot due to a power failure or some other reason we *must* manually reactivate all the tunnels that were active at the time of the failure. After a reboot the tunnels are *not* automatically activated.

After entering all the parameters for our tunnel and confirming this by clicking **OK**, we return to the Virtual Private Network Administration window which shows us all the tunnels we have created. We can select a tunnel and open, to modify, copy, delete, import, export, activate or deactivate it as shown in Figure 352 on page 364.

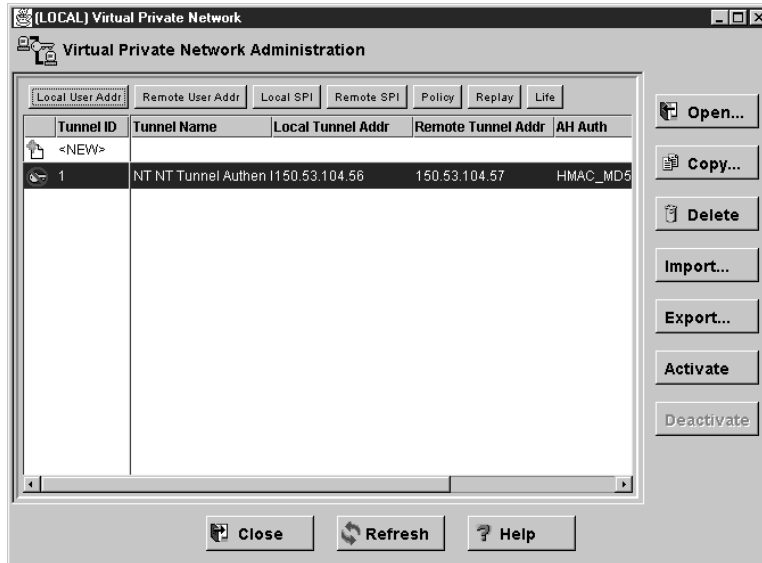


Figure 352. The Tunnel List

15.3.2 Export the Tunnel Definition to a File

We will export the tunnel definition to a file using the GUI. In the tunnel list, (see Figure 352) we select the tunnels we want to export. An *ipsec_tun_man.exp* file will be generated in the specified directory.

Note that since a fixed filename is used for the export file, only one set of tunnel definitions may exist in a given directory at a time. If we are going to have tunnels between different pairs of nodes, we should create different directories for each pair of nodes.

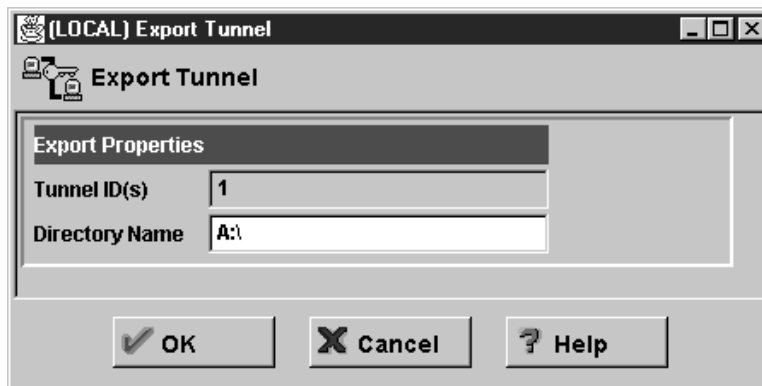


Figure 353. The Export Tunnel Window

When the tunnel configuration export file is created correctly, we should see Figure 354 on page 365.



Figure 354. Successfully Exported Tunnel

Figure 355 shows an example of export file generated by the GUI.

```

Line #-----
1. 4
2. 150.53.104.56
3. 150.53.104.57
4. 1
5. 400
6. 400
7. 256
8. 256
9. NONE
10. 0
11. 0x
12. NONE
13. 0
14. 0x
15. HMAC_MD5
16. 16
17. 0x63aeb6f2d69f8a118e473f2f9680c5ea
18. HMAC_MD5
19. 16
20. 0xd34140e69e179c0e5c4a5285cf38fb91
21. 0
22. 28800
23. tunnel
24. tunnel
25. axax
26. 0
27. 1
28. NONE
29. 0
30.
31. NONE
32. 0
33.
34. 0
35. -
36. -
37. NT NT Tunnel Authen Dynamic
38. 1
39. 9.24.104.242
40. 255.255.255.255
41. 192.168.51.4
42. 255.255.255.255
  
```

Figure 355. Generated Export File

Table 32 on page 366 shows the layout of the export file. When we compare the export file, Figure 355 with the Import File Layout, Table 32 on page 366 we see

on line 38 the value 1 which means a manual tunnel with dynamic filter rules and that we only use authentication, lines 15 through 20.

Table 32 (Page 1 of 2). Tunnel Export - Import File Layout

Line	Field in Export File	Corresponding field in tunnel structure to use when exporting	Corresponding field in tunnel structure to use when importing
1.	IP version number	IP version	IP version
2.	source address	source IP address	destination IP address
3.	destination address	destination IP address	source IP address
4.	tunnel ID	tunnel ID	tunnel ID
5.	dest encr spi > 255	remote esp spi	local esp spi
6.	dest auth spi > 255	remote ah spi	local ah spi
7.	src encr spi > 255	local esp spi	remote esp spi
8.	src auth spi > 255	local ah spi	remote ah spi
9.	receiving encr algorithm	remote esp alg	local esp alg
10.	receiving encr key length	remote esp alg length	local esp alg length
11.	receiving encr key	remote esp key	local esp key
12.	sending encr algorithm	local esp alg	remote esp alg
13.	sending encr key length	local esp alg length	remote esp alg length
14.	sending encr key	local esp key	remote esp key
15.	receiving mac algorithm	remote ah alg	local ah alg
16.	receiving mac key length	remote ah alg length	local ah alg length
17.	receiving mac key	remote ah key	local ah key
18.	sending mac algorithm	local ah alg	remote ah alg
19.	sending mac key length	local ah alg length	remote ah alg length
20.	sending mac key	local ah key	remote ah key
21.	start - defaults to 0	n/a	n/a
22.	time in seconds that the tunnel will be operational	lifetime	lifetime
23.	esp mode - must be tunnel mode	n/a	n/a
24.	ah mode - must be tunnel mode	n/a	n/a
25.	policy	local policy, remote policy	remote policy, local policy
26.	replay (=1), no replay (=0)	replay	replay
27.	new header - must be 1	n/a	n/a
28.	receiving encr mac algorithm	remote enc mac alg	local enc mac alg
29.	receiving encr mac key length	remote enc mac key length	local enc mac key length
30.	receiving encr mac key	remote enc mac key	local enc mac key
31.	sending encr mac algorithm	local enc mac alg	remote enc mac alg
32.	sending encr mac key length	local enc mac key length	remote enc mac key length
33.	sending encr mac key	local enc mac key	remote enc mac key

Table 32 (Page 2 of 2). Tunnel Export - Import File Layout

Line	Field in Export File	Corresponding field in tunnel structure to use when exporting	Corresponding field in tunnel structure to use when importing
34.	through FW	n/a	n/a
35.	FW address	n/a	n/a
36.	destination mask	n/a	n/a
37.	tunnel_name	tunnel name	tunnel name
38.	filter type, static (=0) dynamic (=1)	filter type	filter type
39.	source user address (only dynamic)	local user IP address	remote user IP address
40.	source user mask (only dynamic)	local user mask	remote user mask
41.	dest user address (only dynamic)	remote user IP address	local user IP address
42.	dest user mask (only dynamic)	remote user mask	local user mask

When exporting and importing the tunnel configuration, the owner and partner authentication and encryption keys are switched. The sending key on the owner firewall is the receiving key on the partner.

Note: The SPI's authentication and encryption keys are switched when exporting the tunnel configuration. The source and destination addresses are switched when importing the tunnel configuration on the partner.

15.3.3 Import the Tunnel Definition in the Partner Node

After exporting we have to take the file from the local firewall to the partner firewall. Currently, IBM eNetwork Firewall for Windows NT does not provide any mechanism to do this transfer. The file contains the encryption key for the secure tunnel, so we should devise a secure way to transmit them. We can copy the file on a diskette and go to the partner firewall or we can send the file within an encrypted mail message.

In the partner node, once we have received the files we select **Import** (see Figure 352 on page 364), and we can import the definitions as shown in Figure 356.

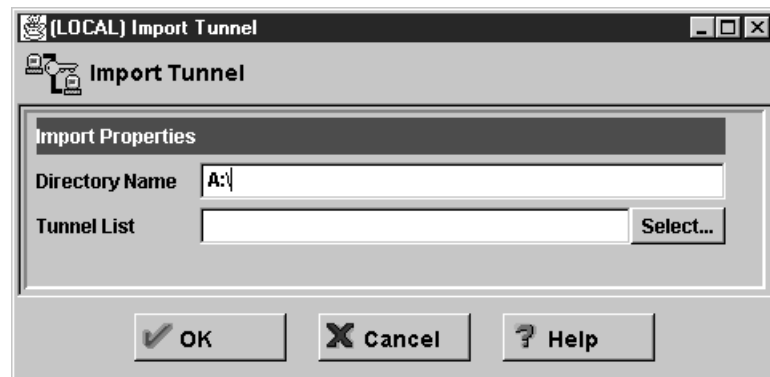


Figure 356. Import Tunnel Window

We enter the directory name and then click on **Select** to get the list of tunnels we can import as shown in Figure 357 on page 368.

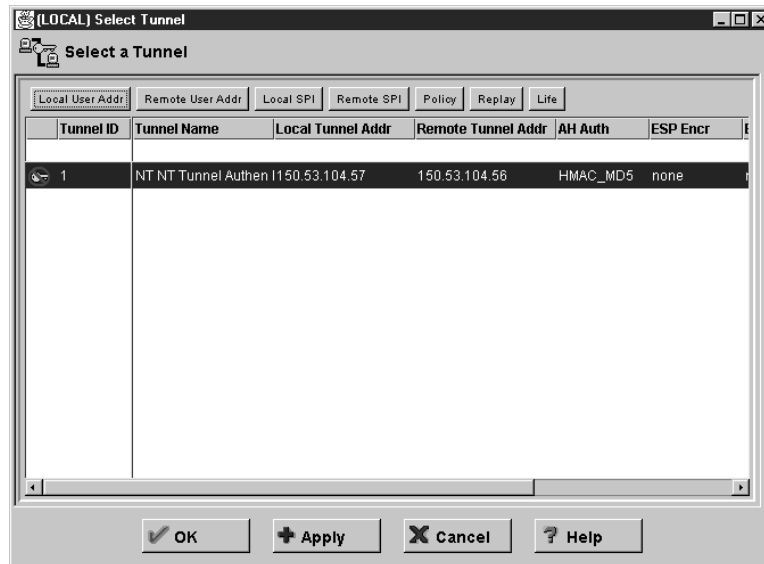


Figure 357. Tunnel Import Selection Window

Once we have selected the tunnels we want to import, we select **OK** and return to the import tunnel window, (Figure 356 on page 367), where we again select **OK** to import our tunnels. When imported successfully we should see Figure 358.

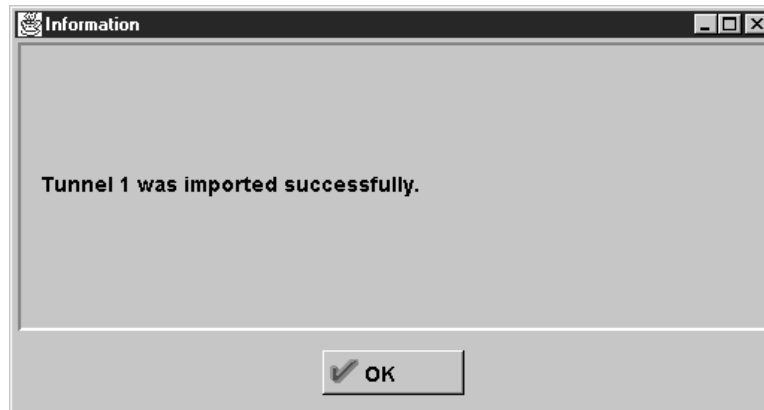


Figure 358. Successfully Imported Tunnel

The import function swaps the source and destination addresses.

Note: The Import function can only be used without modifying the export file if the owner of the tunnel is also an IBM eNetwork Firewall for Windows NT. When the IBM eNetwork Firewall for Windows NT is *not* the owner of the tunnel, the export files on the IBM eNetwork Firewall for Windows NT *must* be updated manually which is not recommended. In the case of a tunnel with an AIX Operating System, the IBM eNetwork Firewall for Windows NT has a utility to do the conversion; see 15.5.2, "Virtual Private Network between IBM eNetwork Firewall for Windows NT and AIX V4.3" on page 391 for details.

15.3.4 Activate/Deactivate the Tunnel at Both Ends

Because we are creating a tunnel with dynamic filter rules, we do not need to create any filter rules or activate the filter rules. We can activate the tunnel at both ends using the GUI. Select the tunnel that you would like to activate and click on **Activate**.

Figure 352 on page 364 shows the tunnel list before activating the tunnel, and Figure 359 shows the active tunnel. You can see that the icon at the left of the tunnel ID is different after the tunnel is activated.

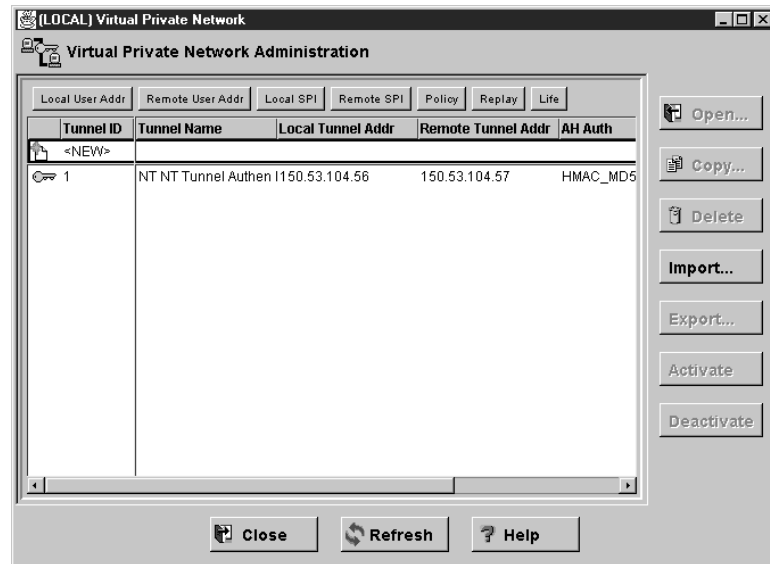


Figure 359. List with Active Tunnel

The tunnel will be marked active even if the other end is not running or connected.

If we want to stop the tunnel, we select it and click on the **Deactivate** button. Every time we activate or deactivate a tunnel message will be written in the log file, (see Figure 360 on page 370).

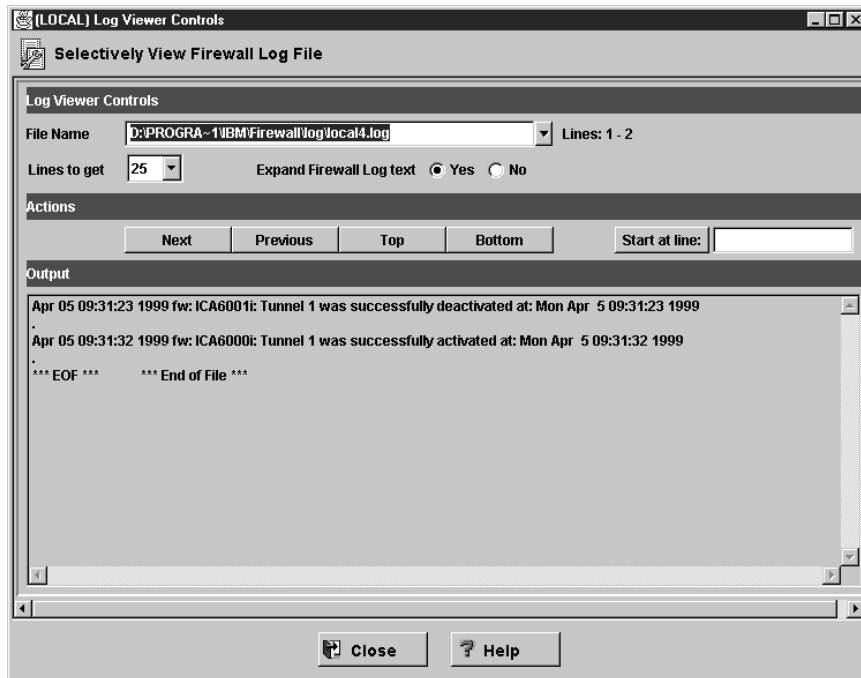


Figure 360. Tunnel Active Logging

When we have executed all the previous steps successfully, we have a tunnel running. When dynamic filters are used, all traffic between the specified users will be transported through the tunnel without regard to the protocol of the traffic. The filter rules are automatically activated.

For example, the filter rules created for the sample configuration of Figure 361 are shown in Figure 362 on page 371.

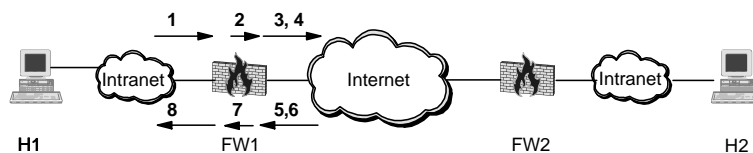


Figure 361. Sample of Manual Tunnel with Dynamic Filter Rules

The numbers match the rules in Figure 362 on page 371.

These are the corresponding filter rules:

```

Rules for a packet from H1 to H2
# Permit inbound packet from H1 to H2
permit H1 H1_MASK H2 H2_MASK all any 0 any 0 both both inbound f=y      1.

# Encapsulate outbound packet from H1 to H2
permit H1 H1_MASK H2 H2_MASK all any 0 any 0 both both outbound f=y t=TID  2.

# Permit encapsulated with AH outbound packet from FW1 to FW2
permit FW1 255.255.255.255 FW2 255.255.255.255 ah any 0 any 0 both both outbound f=y  3.

# Permit encapsulated with ESP outbound packet from FW1 to FW2
permit FW1 255.255.255.255 FW2 255.255.255.255 esp any 0 any 0 both both outbound f=y  4.

Rules for a packet from H2 to H1
# Permit encapsulated with AH inbound packet from FW2 to FW1
permit FW2 255.255.255.255 FW1 255.255.255.255 ah any 0 any 0 both both inbound f=y  5.

# Permit encapsulated with ESP inbound packet from FW2 to FW1
permit FW1 255.255.255.255 FW2 255.255.255.255 esp any 0 any 0 both both inbound f=y  6.

# Permit decapsulated inbound packet from H2 to H1
permit H2 H2_MASK H1 H1_MASK all any 0 any 0 both both inbound f=y t=TID  7.

# Permit outbound packet from H2 to H1
permit H2 H2_MASK H1 H1_MASK all any 0 any 0 both both inbound f=y      8.

```

Figure 362. Generated Dynamic Filter Rules

Where

- FW1 = Nonsecure interface of the local firewall
- FW2 = Nonsecure interface of the partner firewall
- H1 = The local secure client's interface
- H2 = The remote secure client's interface
- H1_MASK, H2_MASK are the subnet masks
- TID = tunnel ID

2. and **7.** are the rules that associate the traffic with the tunnel ID.

Note: H1 and H2 are two specific hosts; we could have also define a subnets instead, and all the hosts in those subnets would be allowed to use the tunnel.

15.3.5 Using Static Filter Rules

By using static filter rules, we can be more specific about which traffic is going to flow through each tunnel. These rules will be like normal rules (with source, target, protocol, ports and port operations), but some of them will also have a tunnel ID. So when a packet must be transferred, the IBM eNetwork Firewall for Windows NT will search the filtering rules. If it matches a rule, and this rule has a specific tunnel ID, the packet will be sent according to the authentication/encryption rules specified in this specific tunnel.

The IBM eNetwork Firewall for Windows NT has provided three services which we can use to define our own rules as shown in Figure 363 on page 372.

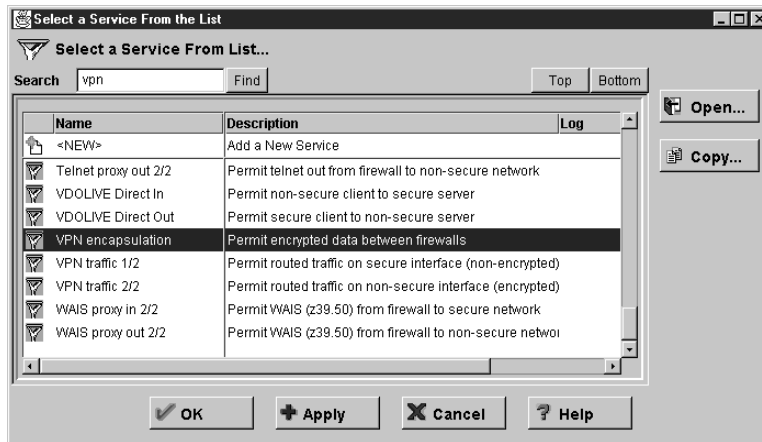


Figure 363. VPN Services

We will create a tunnel with static filter rules with the predefined services. We have to create two connections, one for the encapsulation between the two nonsecure sides of the firewalls and one for the data traffic between the two secure clients as shown in Figure 364.

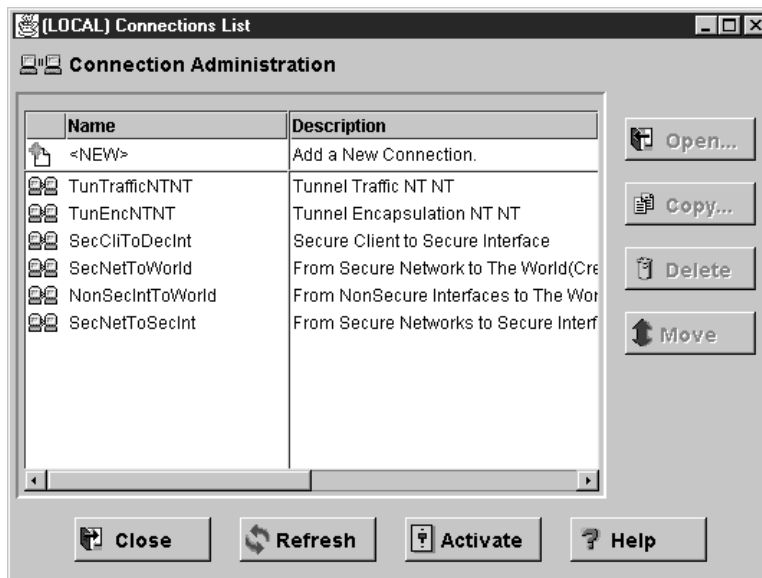


Figure 364. Connection List

The encapsulation takes place between the two nonsecure interfaces of the tunnel partners. For this connection we can always use the predefined service. This will be the same for tunnels with static and dynamic filter rules. We can see the connection setup in Figure 365 on page 373. In Figure 367 on page 375 we can see the four filter rules that are created.

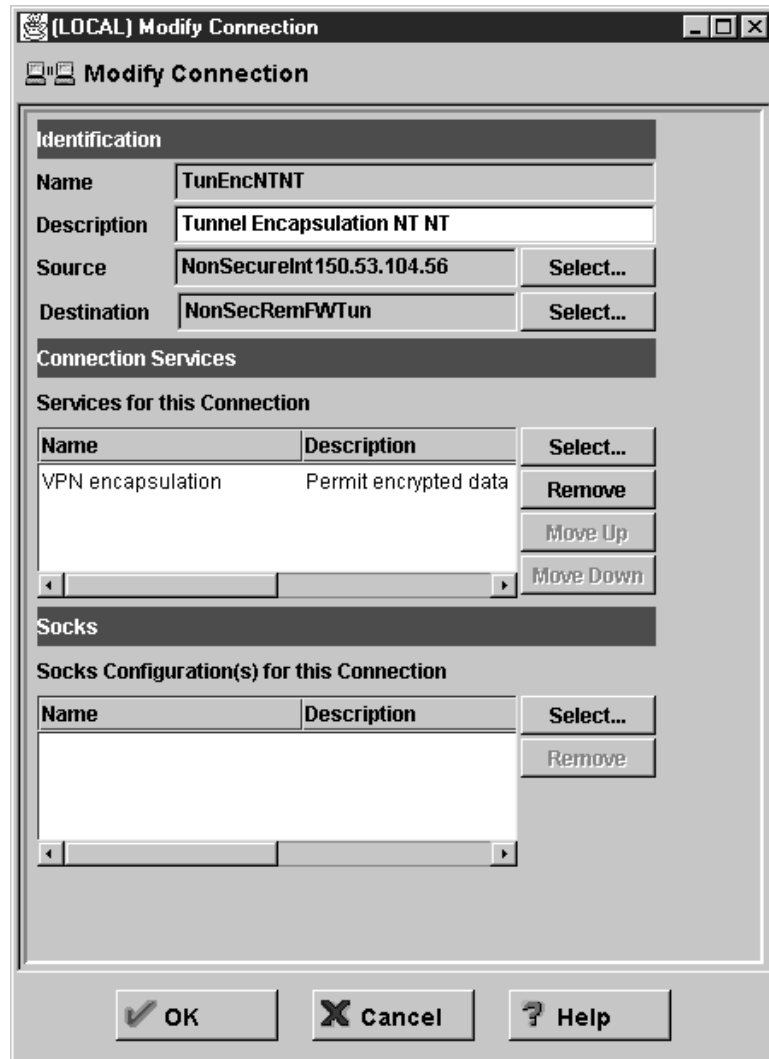


Figure 365. Tunnel Connection Encapsulation

The data traffic connection is between the two clients on each secure side. Here we have two services, one to transport the data from the secure client to the secure interface of the firewall, and one from the nonsecure interface to the tunnel. To get the traffic into the tunnel we need to define the tunnel ID in the second part of the traffic connection. For that reason we cannot use the default service, we *must* copy the VPN traffic 2/2 service into a new service VPN traffic 2/2 Tunnel 3 in our example, and define the tunnel id we want to use. In Figure 377 on page 383 we can see the field **Override Tunnel ID** field where we can select or enter a tunnel ID.

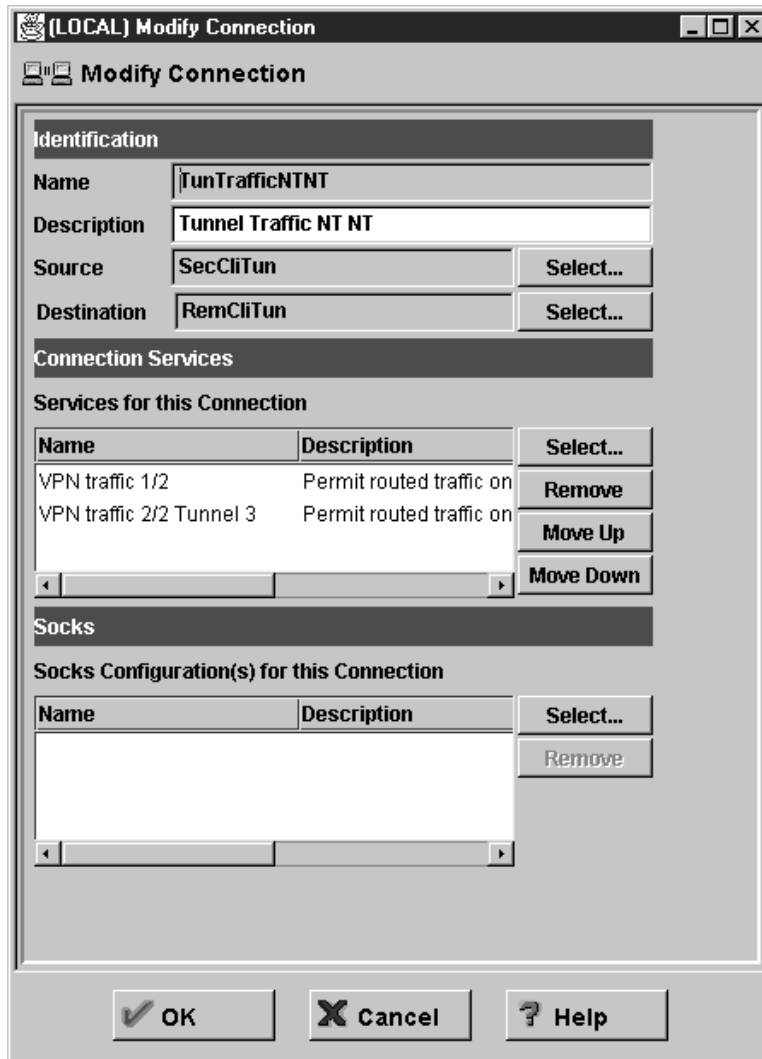


Figure 366. Tunnel Connection Traffic

When we look at Figure 367 on page 375 we see the eight rules that are created. The first two provide the traffic from the secure client to the secure interface of the firewall. The next two are for the transport from the nonsecure interface to the tunnel. At the end of the rules we see that all our traffic will be using Tunnel 3, t=3. The last four rules are for the encapsulation.


```

# Between SecCliTun and RemCliTun
#   Service : VPN traffic 1/2
# Description : Permit routed traffic on secure interface (non-encrypted)
permit 9.24.104.242 255.255.255.255 192.168.51.4 255.255.255.255 all any 0 any 0 secure route inbound l=y f=y
permit 192.168.51.4 255.255.255.255 9.24.104.242 255.255.255.255 all any 0 any 0 secure route outbound l=y f=y
#   Service : VPN traffic 2/2 Tunnel 3
# Description : Permit routed traffic on non-secure interface (encrypted)
permit 9.24.104.242 255.255.255.255 192.168.51.4 255.255.255.255 all any 0 any 0 non-secure route outbound l=y f=y t=3
permit 192.168.51.4 255.255.255.255 9.24.104.242 255.255.255.255 all any 0 any 0 non-secure route inbound l=y f=y t=3
# Between NonSecureInt150.53.104.56 and NonSecRemFWTun
#   Service : VPN encapsulation
# Description : Permit encrypted data between firewalls
permit 150.53.104.56 255.255.255.255 150.53.104.57 255.255.255.255 ah any 0 any 0 non-secure local both l=y f=y
permit 150.53.104.57 255.255.255.255 150.53.104.56 255.255.255.255 ah any 0 any 0 non-secure local both l=y f=y
permit 150.53.104.56 255.255.255.255 150.53.104.57 255.255.255.255 esp any 0 any 0 non-secure local both l=y f=y
permit 150.53.104.57 255.255.255.255 150.53.104.56 255.255.255.255 esp any 0 any 0 non-secure local both l=y f=y

```

Figure 367. Static Rules

This is when we use the default services that come with the IBM eNetwork Firewall for Windows NT. In 15.5, “Virtual Private Network Scenarios” on page 380 we see more detailed examples.

Note: When you are working with VPN tunnels, you should not 'open' the firewall by selecting the Test IP Routing (debug only) option in the Security Policy window. If you do so, the filter rules that direct the traffic to a specific tunnel ID will not be reached, and no traffic will flow through the tunnels.

15.3.6 Reactivate Tunnel When Life Time has Expired

When the tunnel lifetime is reached, the tunnel will cease operation until it is reactivated. Figure 352 on page 364 shows the VPN Administration screen with an expired manual tunnel.

To reactivate the tunnel, simply select the tunnel and click on the **Activate** button. However, reactivating a tunnel would only reactivate it to an operational state. The keys used in the tunnel remain the same. To re-establish the tunnel with new session keys, you need to delete the tunnel, and then add a new tunnel with the same tunnel ID and characteristics back into the firewall. After that, you export the new tunnel definition to the tunnel partner. New session keys are stored inside the definition file. Your tunnel partner is also required to delete the existing tunnel and then re-import the new definition.

15.3.7 Summary

The following list is a summary of the steps to create and activate a tunnel:

1. Create a firewall object for the nonsecure interface of the remote firewall.
2. Create a network object for the secure network of the remote firewall (or for the specific hosts with which we want to be connected).
3. Create the tunnel itself; local address=nonsecure interface of the local firewall and remote address=remote firewall object from 1).
4. Export definitions, transport to the remote firewall, import definitions (which automatically switches local and remote addresses).
5. Add connection services for both VPN encapsulation; source=nonsecure interface of my firewall, destination=remote firewall object from 1. This allows firewall-to-firewall communication of encapsulated data.

6. Copy the VPN 2/2 rule to a new rule called "VPN traffic 2/2 tunnel xx" and set the tunnel ID in that rule.
7. Create a connection service "VPN traffic 1/2" and "VPN traffic 2/2 tunnel xx" from 6. Source=my secure network (or the set of hosts allowed to use VPN), destination=secure network of remote firewall from 2).
8. Ensure IPforwarding is set to "ON"; see 4.6.3, "Activating Windows NT's IP Forwarding Function" on page 57.
9. Repeat steps 1, 2, 5, 6, 7 and 8 at the remote firewall. (Note that now "remote" and "local" are relative to that firewall.)
10. Activate rule sets.
11. Activate the tunnel at both ends.
12. Try to ping between the networks; it should work!

15.4 Authentication and Encryption Examples

In order to understand how the tunnels work, we will show two examples, both using the same tunnel, in the network shown in Figure 374 on page 381. Both firewalls are configured to allow routed FTP traffic using the default services VPN Traffic 1/2 and VPN Traffic 2/2. We made an FTP connection from client 9.24.104.242 to FTP server 192.168.51.4 and executed the DIR command, which gave us the output in Figure 368. In the first example we use only authentication and in the second authentication and encryption. We compared the data packet that was transported from the FTP server to the client.

```

C:\>ftp 192.168.51.4
Connected to 192.168.51.4.
220 home Microsoft FTP Service (Version 3.0).
User (192.168.51.4:(none)): anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password:
230>Welcome to this FTP site hosting:

- ftp.securez.com
- ftp.faire.com

Enjoy!
230 Anonymous user logged in.
ftp> dir
200 PORT command successful.
150 Opening ASCII mode data connection for /bin/ls.
----- 1 owner group 18651694 Dec 11 1998 cp32d45.exe
d----- 1 owner group 0 Sep 15 1998 I386
----- 1 owner group 65307272 Apr 2 19:46 I386.zip
d----- 1 owner group 0 Mar 29 22:12 ie401
----- 1 owner group 23910912 Mar 29 22:16 ie401.exe
d----- 1 owner group 0 Sep 15 1998 serupak3
d----- 1 owner group 0 Mar 25 16:54 TR-ISA
----- 1 owner group 87570195 Apr 1 16:48 WebSphere.zip
----- 1 owner group 695808 Apr 2 19:45 Winzip.exe
226 Transfer complete.
623 bytes received in 0.01 seconds (41.53 Kbytes/sec)
ftp> -

```

Figure 368. FTP Download

Before using the tunnel we took a trace of the FTP data packet that was sent to the client. This packet will be compared with the packets in each example. Figure 369 on page 377 shows us the data packet not using a tunnel.

```

IP: Source Address = 192.168.51.4
IP: Destination Address = 9.24.104.242
IP: Data: Number of data bytes remaining = 643 (0x0283)
00000000 10 40 00 20 35 03 E7 10 08 00 5A CE 6D 77 AA AA .@. 5.+...Z+mw~
00000010 03 00 00 00 08 00 45 00 02 97 D0 1E 40 00 80 06 .....@..ù-@.Ç.
00000020 C2 8E C0 A8 33 04 09 18 68 F2 00 14 07 D4 0E 88 -Å+ç3...h=Ŧ.+ê
00000030 EE D4 22 BA E7 E9 50 18 3E A8 E0 FD 00 00 2D 2D e+";tTP.>çã²...
00000040 2D 2D 2D 2D 2D 2D 2D 2D 20 20 20 31 20 6F 77 6E ----- l own
00000050 65 72 20 20 20 20 67 72 6F 75 70 20 20 20 20 20 er group
00000060 20 20 20 31 38 36 35 31 36 39 34 20 44 65 63 20 18651694 Dec
00000070 31 31 20 20 31 39 39 38 20 63 70 33 32 64 34 35 11 1998 cp32d45
00000080 2E 65 78 65 0D 0A 64 2D 2D 2D 2D 2D 2D 2D 2D .exe..d-----
00000090 20 20 20 31 20 6F 77 6E 65 72 20 20 20 20 67 72 l owner gr
000000A0 6F 75 70 20 20 20 20 20 20 20 20 20 20 20 20 20 oup
000000B0 20 20 30 20 53 65 70 20 31 35 20 20 31 39 39 38 0 Sep 15 1998
000000C0 20 69 33 38 36 0D 0A 2D 2D 2D 2D 2D 2D 2D 2D 2D i386..-----

```

Figure 369. FTP Data Packet (No Tunnel)

In the heading we see that the data sent is 0x0283 (643) bytes. We have positioned the cursor on the first byte of the IP packet that contains the IP version. The third byte on line 20 gives us our source address C0 A8 33 04 (192.168.51.4), immediately followed by the destination address 09 18 68 72 (9.24.104.242). Next comes the TCP header with source port 00 14 (20, FTP data) and the destination port 07 D4 (2004 ephemeral). The data starts on line 30, the two last bytes with 2D 2D. In the right column we see the first two lines of data of Figure 368 on page 376.

15.4.1 Authentication Example

In this first example we send the packet through the tunnel using authentication only. To get a better view we have traced the packet before (see Figure 370) and after (see Figure 371 on page 378) it was authenticated.

```

IP: Source Address = 192.168.51.4
IP: Destination Address = 9.24.104.242
IP: Data: Number of data bytes remaining = 643 (0x0283)
00000000 10 40 00 20 35 03 E7 10 08 00 5A CE 6D 77 AA AA .@. 5.+...Z+mw~
00000010 03 00 00 00 08 00 45 00 02 97 D1 0E 40 00 80 06 .....@..ù-@.Ç.
00000020 C1 9B C0 A8 33 04 09 18 68 F2 00 14 04 D5 08 6A -ç+ç3...h=Ŧ.+j
00000030 33 F7 08 7E 15 9E 50 18 3E A8 91 83 00 00 2D 2D 3".{$PP.>çãâ...
00000040 2D 2D 2D 2D 2D 2D 2D 2D 20 20 20 31 20 6F 77 6E ----- l own
00000050 65 72 20 20 20 20 67 72 6F 75 70 20 20 20 20 20 er group
00000060 20 20 20 31 38 36 35 31 36 39 34 20 44 65 63 20 18651694 Dec
00000070 31 31 20 20 31 39 39 38 20 63 70 33 32 64 34 35 11 1998 cp32d45
00000080 2E 65 78 65 0D 0A 64 2D 2D 2D 2D 2D 2D 2D 2D .exe..d-----
00000090 20 20 20 31 20 6F 77 6E 65 72 20 20 20 20 67 72 l owner gr
000000A0 6F 75 70 20 20 20 20 20 20 20 20 20 20 20 20 20 oup
000000B0 20 20 30 20 53 65 70 20 31 35 20 20 31 39 39 38 0 Sep 15 1998
000000C0 20 69 33 38 36 0D 0A 2D 2D 2D 2D 2D 2D 2D 2D 2D i386..-----

```

Figure 370. FTP Data Packet before Authentication

When we compare the packet in Figure 370 before authentication with the packet in Figure 369 we see there is no difference. The source and destination address are still the same and the data sent is still 0x0283 (643) bytes, no header has been added.

```

IP: Source Address = 150.53.104.57
IP: Destination Address = 150.53.104.56
IP: Data: Number of data bytes remaining = 687 (0x02AF)

```

00000000	18 40 08 00 5A B9 69 05 08 00 5A B9 68 75 AA AA	.@..Z i...Z;hu--
00000010	03 00 00 00 08 00 45 00 02 C3 4C 00 00 00 7E 33+L....~3
00000020	F1 2B 96 35 68 39 96 35 68 38 04 04 00 00 00 00	++û5h9û5h8.....
00000030	01 00 00 00 00 2A 3B A3 72 E2 8B 4D 6E D8 1D F2*;ûr;iMn+=
00000040	90 C8 45 00 02 97 D1 0E 40 00 7F 06 C2 9B C0 A8	Ë+E..ù-.@.D.-+;ç
00000050	33 04 09 18 68 F2 00 14 04 D5 08 6A 33 F7 08 7B	3...h=.Ŧ.+j3".{
00000060	15 9E 50 18 3E A8 91 83 00 00 2D 2D 2D 2D 2D 2D	\$PP.>çæâ.-----
00000070	2D 2D 2D 2D 20 20 20 31 20 6F 77 6E 65 72 20 20	---- 1 owner
00000080	20 20 67 72 6F 75 70 20 20 20 20 20 20 20 31	group 1
00000090	38 36 35 31 36 39 34 20 44 65 63 20 31 31 20 20	8651694 Dec 11
000000A0	31 39 39 38 20 63 70 33 32 64 34 35 2E 65 78 65	1998 cp32d45.exe
000000B0	0D 0A 64 2D 2D 2D 2D 2D 2D 2D 2D 2D 20 20 20 31	..d----- 1
000000C0	20 6F 77 6E 65 72 20 20 20 20 67 72 6F 75 70 20	owner group
000000D0	20 20 20 20 20 20 20 20 20 20 20 20 20 30 20	0
000000E0	53 65 70 20 31 35 20 20 20 31 39 39 38 20 69 33 38	Sep 15 1998 i38
000000F0	36 0D 0A 2D 2D 2D 2D 2D 2D 2D 2D 2D 20 20 20	6..-----

Figure 371. FTP Data Packet after Authentication

When we look at the packet after authentication (see Figure 371) and compare it with the packet before authentication (see Figure 370 on page 377) we see that the source and destination addresses have changed. The source address, third byte on line 20, 96 35 68 39 is now the address of the nonsecure interface of the sending firewall FW2 (150.53.104.57) and the destination address, 96 35 68 38, is the address of the nonsecure interface (150.53.104.56) of the receiving firewall (FW1).

The original source address is now on line 40, the two last bytes and the first two bytes of line 50, followed by the original destination address, bytes three through six. The third byte on line 40, value 45, is the first byte of the original packet. The size of the data packet is no longer 643 but 687 bytes. Between the new IP header and the original IP header a new AH header is added. This new header starts at byte 11 on line 20, value 04, and ends with the second byte on line 40, value C8. This new added header is the authentication header. The format of the authentication header can be found in RFC 2402. One of the values in the authentication header is the Security Parameter Index (SPI). The SPI is 32 bits long, the last two bytes of line 20 and the first two bytes of line 30, which gives us value 00 00 01 00 (HEX), or 256 in decimal. When creating the tunnel the local SPI was automatically set to 256; when transporting packets the authentication header always contains the local SPI of the receiving tunnel endpoint.

The original IP packet is encapsulated with a new IP header containing the addresses of the two endpoints of our tunnel, and an authentication header is added.

15.4.2 Encryption Example

For the encryption example we will execute exactly the same FTP command as we did for the authentication, but now with encryption using both AH and ESP.

```

IP: Source Address = 192.168.51.4
IP: Destination Address = 9.24.104.242
IP: Data: Number of data bytes remaining = 643 (0x0283)

```

00000000	10 40 00 20 35 03 B7 10 08 00 5A CE 6D 77 AA AA	.@. 5+...Z+mw---
00000010	03 00 00 00 08 00 45 00 02 97 29 0F 40 00 80 06@..ù)æ@.Ç.
00000020	69 9B C0 A8 33 04 09 18 68 F2 00 14 04 DC 08 84	ic+ç3...h=.Œ._.ã
00000030	EB FF 08 95 CC 6D 50 18 3E A8 22 70 00 00 2D 2D	d .ò;mP.>ç"p...--
00000040	2D 2D 2D 2D 2D 2D 2D 2D 20 20 20 31 20 6F 77 6E	----- l own
00000050	65 72 20 20 20 20 67 72 6F 75 70 20 20 20 20 20	er group
00000060	20 20 20 31 38 36 35 31 36 39 34 20 44 65 63 20	18651694 Dec
00000070	31 31 20 20 31 39 39 38 20 63 70 33 32 64 34 35	ll 1998 cp32d45
00000080	2E 65 78 65 0D 0A 64 2D 2D 2D 2D 2D 2D 2D 2D	.exe..d-----
00000090	20 20 20 31 20 6F 77 6E 65 72 20 20 20 20 67 72	l owner gr
000000A0	6F 75 70 20 20 20 20 20 20 20 20 20 20 20 20 20	oup
000000B0	20 20 30 20 53 65 70 20 31 35 20 20 31 39 39 38	0 Sep 15 1998
000000C0	20 69 33 38 36 0D 0A 2D 2D 2D 2D 2D 2D 2D 2D	i386..-----

Figure 372. FTP Data Packet before Encryption

When we compare the packet in Figure 372, the packet before encryption, with the packet in Figure 369 on page 377, using no tunnel, and with the packet in Figure 370 on page 377 before authentication, we see no difference. The source address is the third byte on line 20 with value C0 A8 33 04 (192.168.51.4) and it continues with the destination address 09 18 68 F2 (9.24.104.242). The destination address is followed by the TCP header and the data starts at the second to last byte of line 30 with 2D 2D.

Figure 373 on page 380 shows the packet after encryption. The size of the data packet is now 0x02D4 (724). The packet is encapsulated with a new IP header and contains the two endpoints of the tunnel. The third byte on line 20, 96 35 68 39 (150.53.104.57), is the source address followed by the destination address, 96 35 68 38 (150.53.104.56). These are the two tunnel endpoints. All the headers and the data that comes after the new IP header is encrypted as we can see in the right column. The original source and destination are no longer readable nor is the data that we are transporting.

```

IP: Source Address = 150.53.104.57
IP: Destination Address = 150.53.104.56
IP: Data: Number of data bytes remaining = 724 (0x02D4)

```

00000010	03 00 00 00 08 00 45 00 02 E8 6C 00 00 00 7E 33E..F1...~3
00000020	D1 06 96 35 68 39 96 35 68 38 32 04 00 00 00 00	-.û5h9û5h82.....
00000030	01 01 00 00 00 1A E1 DD ED 57 D6 D5 9B C3 A5 23B;+W++c+M#
00000040	87 E8 00 00 01 01 00 00 00 1A 00 00 00 00 1E 00	ç+.....
00000050	00 00 75 96 23 82 9C C4 64 5D FE 17 F0 C8 67 8B	..uû#éZ-d .=+gi
00000060	85 49 DF EB E6 80 77 60 7F D7 3E F8 CB 77 74 9B	àI`dµçw`O+>"-wtc
00000070	51 27 0A CB D1 46 47 29 FE 77 76 72 D0 BC 7C 23	Q'--FG) wvr-+ #
00000080	30 1E AF CE 05 08 DC FE 7D 4D 3D 20 C3 73 98 AA	00>+..._)M= +sÿ~
00000090	6D 7A 06 07 B9 EB CE 5E 28 23 AE 49 06 AB F1 B5	mz..;+^^{#I.±±!
000000A0	83 05 BF E6 8B 89 66 DE 3E 13 7C DD 2F 6C 16 CE	â.+:!ëf!>. /1.+
000000B0	23 A4 65 46 61 22 1A 4F 5B 7E C0 E3 80 60 AD BC	#ñeFa".0[~+pç`+;
000000C0	AE 37 FF 49 83 C9 F3 0F B9 07 C8 AE F8 CC A7 00	±7 Iâ+±±.±±±!°.
000000D0	4E 60 ED D1 CE E0 CE 83 58 7C 6A 65 6D 83 11 93	K`f--a+âX jemâ.ô
000000E0	4C 23 EE 13 D6 AD 9C CC 70 1E 9D 02 57 36 30 3B	L#e.+:; pOÿ.W60;
000000F0	26 EE E5 A3 81 36 B1 FC F0 37 E8 74 1E 17 1D AC	æesüü6;n=7FtO...4
00000100	A8 33 28 19 D3 C6 5D 24 34 9F 89 10 04 59 2F FC	ç3(.+ \$4fë..Y/n
00000110	7A 65 C3 11 63 F2 0B 85 37 DD 18 63 20 D6 2C 01	ze+.c.=.â7;.c +,.
00000120	6D 4D 5B 5F B9 E1 54 94 35 2B 3C F8 97 22 D7 5B	mM[...;Tô5+<"ù"+
00000130	36 4F 69 09 25 2D 82 42 66 C8 3E EB C1 9D 6D 06	60i.±-éBf+>d-ÿm.
00000140	9D 71 D5 7E 3D 5E C8 A9 E9 18 F2 97 C5 66 29 EC	ÿqt~+{+T.=û+f)8
00000150	A0 34 92 CE AB C0 58 AA FF 12 1D 62 58 96 BE 13	â4E+±+X~...bXû+.
00000160	86 2B 15 19 CF 78 8E 06 ED 67 13 42 08 4C AB E6	â+\$.-xâ.fg.B.L±µ
00000170	7B 51 72 6D 3B EC A9 DB E8 5F BC 50 41 14 C3 6A	{Qr;8~; F_+PA[+j
00000180	C4 86 3C 95 BE 9D 9E 8A F9 2C 6E E5 60 70 52 07	-â<ô+ÿPè~,ns`pR.
00000190	BA AA 8B 00 E1 EE C5 D3 5D A7 05 55 EA 68 23 5F	-i.Be++ °.U0h#_

Figure 373. Packet after Encryption

15.5 Virtual Private Network Scenarios

We have configured the following tunnels :

1. Between two IBM eNetwork Firewall for Windows NTs using static filter rules
2. Between a IBM eNetwork Firewall for Windows NT and the AIX V4.3 operating system using dynamic filter rules.
3. Between an IBM eNetwork Firewall for Windows NT and the AS/400 V4R4 using the default static filter rules VPN Traffic 1/2 and VPN Traffic 2/2
4. Between an IBM eNetwork Firewall for Windows NT and the OS/390 Firewall Technologies V2.8 beta using static filter rules.

In all the configurations we tested, we created the traffic rules on the partner tunnel manually.

When configuring a tunnel with another system different from the IBM eNetwork Firewall for Windows NT, you must be sure that the authentication or encryption method you want to use is supported.

15.5.1 Tunnel between Two IBM eNetwork Firewalls for Windows NT

In this scenario we will configure two tunnels static filter rules between our two secure networks over a nonsecure network, allowing only FTP and Telnet, each using its own tunnel.

The test network has two firewalls, FW1 and FW2, protecting the secure networks 9.24.104.0 and 192.168.51.0. Their nonsecure IP addresses are respectively 150.53.104.56 and 150.53.104.57 as shown in Figure 374 on page 381.

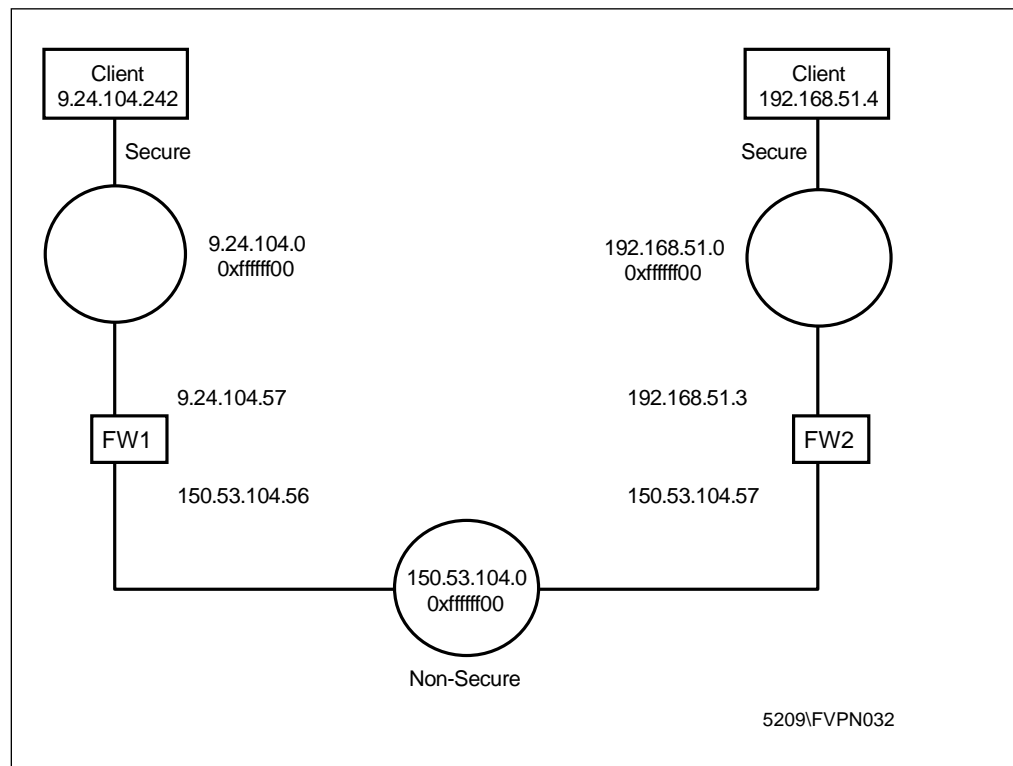


Figure 374. Network Tunnel Configuration between Two IBM eNetwork Firewall for Windows NT

Tunnel 3 will be used for Telnet and tunnel 4 for FTP. The secure Telnet and FTP server are both on the same host and have 192.168.51.4 as IP address. On the other secure network we have our secure client with 9.24.104.242 as the IP address. For Telnet traffic, the packets will be sent using authentication and encryption and for FTP traffic we use only encryption. Figure 375 shows the tunnel configuration.

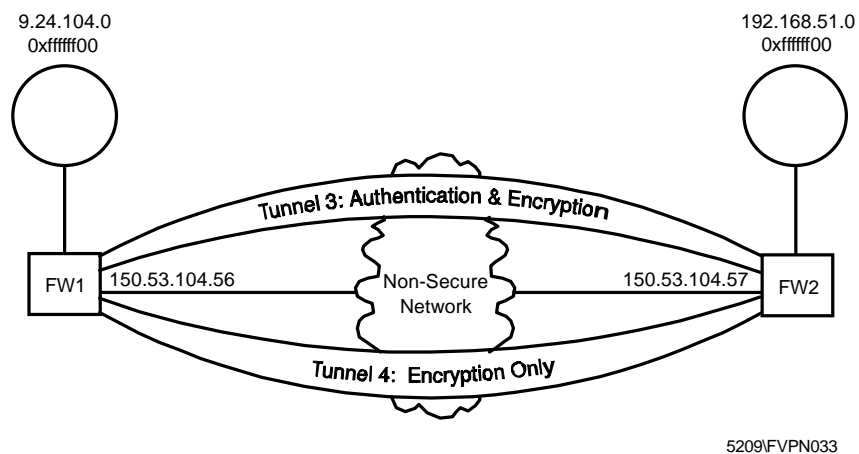


Figure 375. Tunnel Configuration for Testing

Following the procedure we described earlier, we first defined the tunnels in FW1, exported the definitions, moved them to FW2 and imported them there. The next step is to create the connections, one for the encapsulation, the other for the traffic, as shown in Figure 364 on page 372. For the encapsulation we can use the default service, as shown in Figure 365 on page 373. For the traffic we need to configure two services, for both protocols. The first service is for the traffic to the secure interface and the second from the nonsecure interface to the tunnel. As we can see in Figure 376.

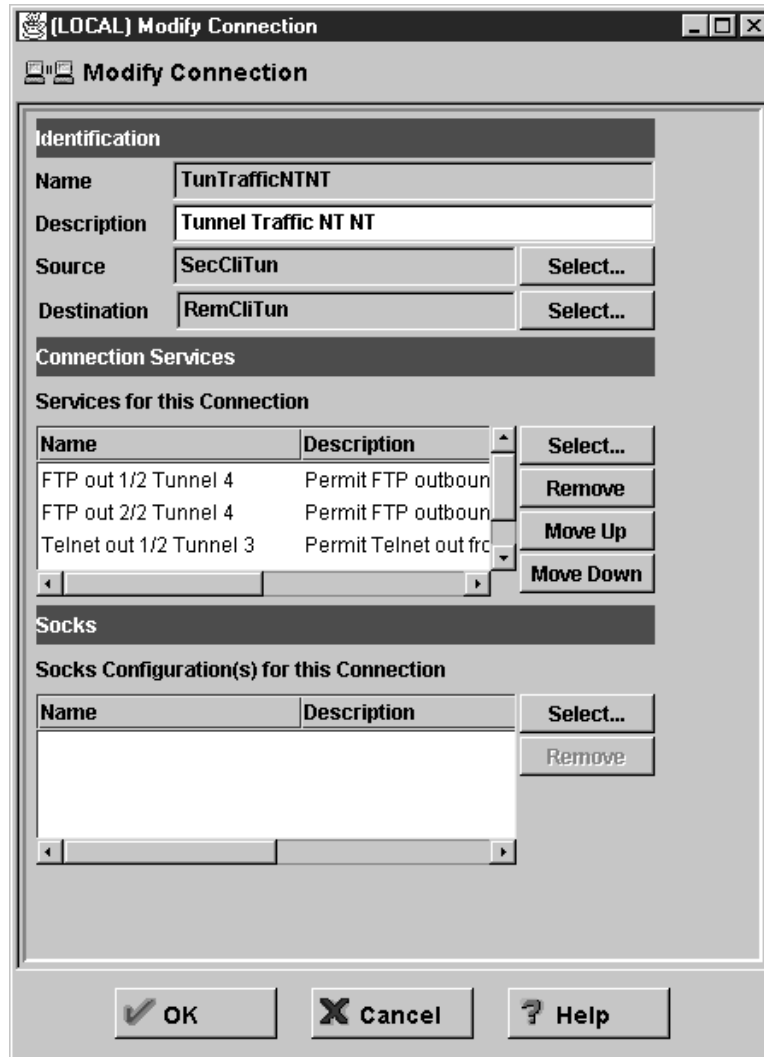


Figure 376. Tunnel Traffic Connection

At the service level, we will define which protocol will use which tunnel. Figure 377 on page 383 shows us that Tunnel 4 will be used for FTP and for Telnet we do exactly the same, now using Tunnel 3.

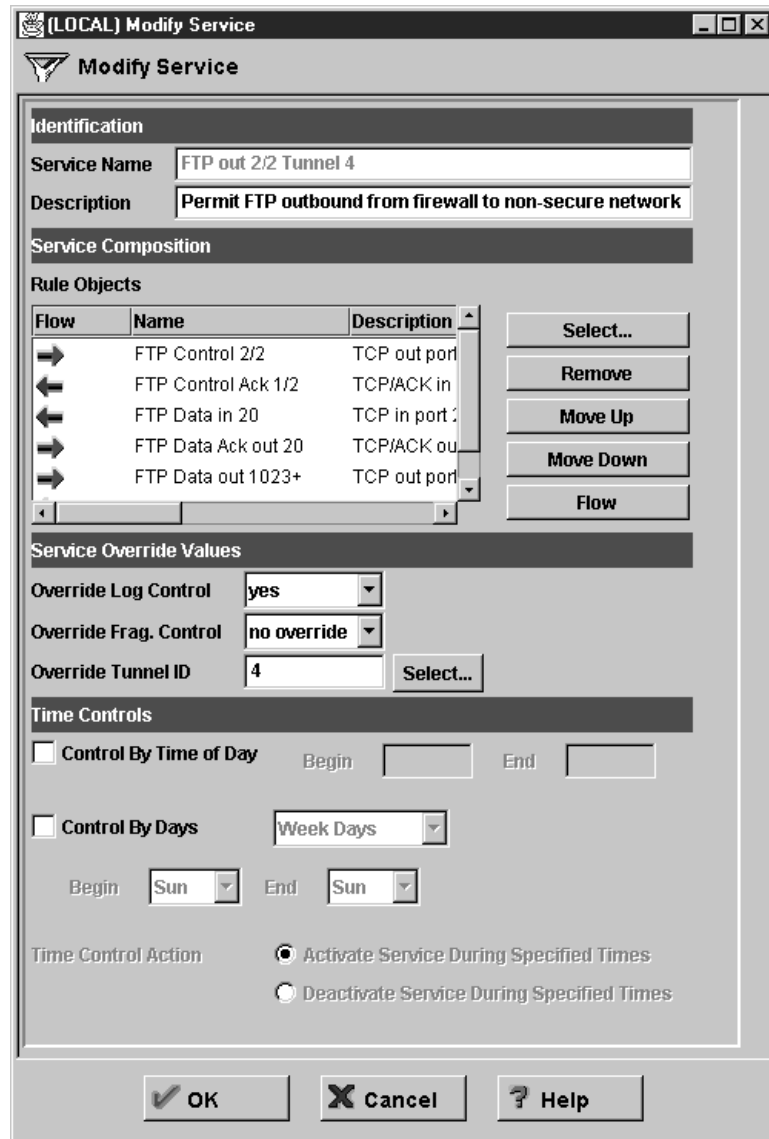


Figure 377. FTP Traffic Service with Tunnel Definition

The tunnel can also be defined at the rule level because all FTP traffic uses only one tunnel and all Telnet traffic also uses only one tunnel we can do it at the service level.

For each tunnel, packets come in the clear from the secure side, then they are sent through one of the tunnels, received at the second end of the tunnel and go again in the clear to the final destination. Figure 378 on page 384 shows the filter rules that we used in FW1 for the tunnels.

```

#      Service : FTP out 1/2 Tunnel 4
# Description : Permit FTP outbound from secure network to firewall
permit 9.24.104.242 255.255.255.255 192.168.51.4 255.255.255.255 tcp gt 1023 eq 21 secure route inbound l=y f=y
permit 192.168.51.4 255.255.255.255 9.24.104.242 255.255.255.255 tcp/ack eq 21 gt 1023 secure route outbound l=y f=y
permit 192.168.51.4 255.255.255.255 9.24.104.242 255.255.255.255 tcp eq 20 gt 1023 secure route outbound l=y f=y
permit 9.24.104.242 255.255.255.255 192.168.51.4 255.255.255.255 tcp/ack gt 1023 eq 20 secure route inbound l=y f=y
permit 9.24.104.242 255.255.255.255 192.168.51.4 255.255.255.255 tcp gt 1023 gt 1023 secure route inbound l=y f=y
permit 192.168.51.4 255.255.255.255 9.24.104.242 255.255.255.255 tcp/ack gt 1023 gt 1023 secure route outbound l=y f=y
#      Service : FTP out 2/2 Tunnel 4
# Description : Permit FTP outbound from firewall to non-secure network
permit 9.24.104.242 255.255.255.255 192.168.51.4 255.255.255.255 tcp gt 1023 eq 21 non-secure route outbound l=y f=y t=4
permit 192.168.51.4 255.255.255.255 9.24.104.242 255.255.255.255 tcp/ack eq 21 gt 1023 non-secure route inbound l=y f=y t=4
permit 192.168.51.4 255.255.255.255 9.24.104.242 255.255.255.255 tcp eq 20 gt 1023 non-secure route inbound l=y f=y t=4
permit 9.24.104.242 255.255.255.255 192.168.51.4 255.255.255.255 tcp/ack gt 1023 eq 20 non-secure route outbound l=y f=y t=4
permit 9.24.104.242 255.255.255.255 192.168.51.4 255.255.255.255 tcp gt 1023 gt 1023 non-secure route outbound l=y f=y t=4
permit 192.168.51.4 255.255.255.255 9.24.104.242 255.255.255.255 tcp/ack gt 1023 gt 1023 non-secure route inbound l=y f=y t=4
#      Service : Telnet out 1/2
# Description : Permit Telnet out from secure network to firewall
permit 9.24.104.242 255.255.255.255 192.168.51.4 255.255.255.255 tcp gt 1023 eq 23 secure route inbound l=y f=y
permit 192.168.51.4 255.255.255.255 9.24.104.242 255.255.255.255 tcp/ack eq 23 gt 1023 secure route outbound l=y f=y
#      Service : Telnet out 2/2
# Description : Permit telnet out from firewall to non-secure network
permit 9.24.104.242 255.255.255.255 192.168.51.4 255.255.255.255 tcp gt 1023 eq 23 non-secure route outbound l=y f=y t=3
permit 192.168.51.4 255.255.255.255 9.24.104.242 255.255.255.255 tcp/ack eq 23 gt 1023 non-secure route inbound l=y f=y t=3
# Between NonSecureInt150.53.104.56 and NonSecRemFWTun
#      Service : VPN encapsulation
# Description : Permit encrypted data between firewalls
permit 150.53.104.56 255.255.255.255 150.53.104.57 255.255.255.255 ah any 0 any 0 non-secure local both l=y f=y
permit 150.53.104.57 255.255.255.255 150.53.104.56 255.255.255.255 ah any 0 any 0 non-secure local both l=y f=y
permit 150.53.104.56 255.255.255.255 150.53.104.57 255.255.255.255 esp any 0 any 0 non-secure local both l=y f=y
permit 150.53.104.57 255.255.255.255 150.53.104.56 255.255.255.255 esp any 0 any 0 non-secure local both l=y f=y

```

Figure 378. Tunnel Filter Rules on Firewall 1 (FW1)

The first six rules are for FTP from the secure network to the secure interface of the firewall (FW1). The second part of the rules is the FTP traffic from the nonsecure interface to the nonsecure network. In these rules at the end of the line the tunnel ID is added, all the traffic for ports 21 and 20 (FTP) will use tunnel 4. The third part of rules are the Telnet rules from the secure network to the secure interface of the firewall (FW1). The fourth part of the rules is from the nonsecure firewall interface to the nonsecure network, defining that we have to use tunnel 3 for port 23 (Telnet). The last part of the rules allows the AH and ESP protocols between the firewalls to authenticate and encrypt the packets.

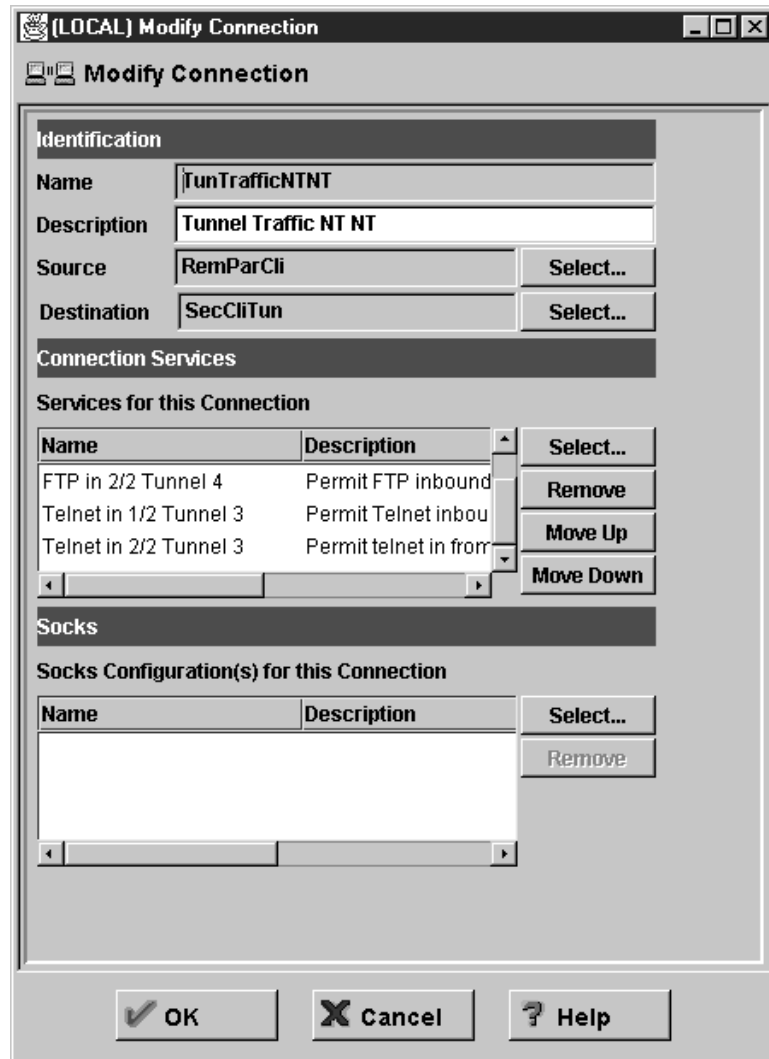


Figure 379. Tunnel Traffic Connection at Partner End.

On the partner end of the tunnel we have to configure the same connections as we did on the owner end. The only difference is that we have to allow an inbound connection. We create the inbound services for FTP and Telnet as shown in Figure 379. We allow only Telnet traffic from Tunnel 3 and FTP traffic from Tunnel 4. We need to specify the tunnel ID in the rules that allow traffic from the nonsecure side of the owner firewall to the nonsecure interface of the firewall. (see Figure 380 on page 386). Figure 379 shows the connection at the partner end.

```

# Between RemParCli and SecCliTun
# Service : FTP in 1/2 Tunnel 4
# Description : Permit FTP inbound from non-secure network to firewall
permit 9.24.104.242 255.255.255.255 192.168.51.4 255.255.255.255 tcp gt 1023 eq 21 non-secure route inbound l=y f=y t=4
permit 192.168.51.4 255.255.255.255 9.24.104.242 255.255.255.255 tcp/ack eq 21 gt 1023 non-secure route outbound l=y f=y t=4
permit 192.168.51.4 255.255.255.255 9.24.104.242 255.255.255.255 tcp eq 20 gt 1023 non-secure route outbound l=y f=y t=4
permit 9.24.104.242 255.255.255.255 192.168.51.4 255.255.255.255 tcp/ack gt 1023 eq 20 non-secure route inbound l=y f=y t=4
permit 9.24.104.242 255.255.255.255 192.168.51.4 255.255.255.255 tcp gt 1023 gt 1023 non-secure route inbound l=y f=y t=4
permit 192.168.51.4 255.255.255.255 9.24.104.242 255.255.255.255 tcp/ack gt 1023 gt 1023 non-secure route outbound l=y f=y t=4
# Service : FTP in 2/2 Tunnel 4
# Description : Permit FTP inbound from firewall to secure network
permit 9.24.104.242 255.255.255.255 192.168.51.4 255.255.255.255 tcp gt 1023 eq 21 secure route outbound l=y f=y
permit 192.168.51.4 255.255.255.255 9.24.104.242 255.255.255.255 tcp/ack eq 21 gt 1023 secure route inbound l=y f=y
permit 192.168.51.4 255.255.255.255 9.24.104.242 255.255.255.255 tcp eq 20 gt 1023 secure route inbound l=y f=y
permit 9.24.104.242 255.255.255.255 192.168.51.4 255.255.255.255 tcp/ack gt 1023 eq 20 secure route outbound l=y f=y
permit 9.24.104.242 255.255.255.255 192.168.51.4 255.255.255.255 tcp gt 1023 gt 1023 secure route outbound l=y f=y
permit 192.168.51.4 255.255.255.255 9.24.104.242 255.255.255.255 tcp/ack gt 1023 gt 1023 secure route inbound l=y f=y
# Service : Telnet in 1/2
# Description : Permit Telnet inbound from non-secure network to firewall
permit 9.24.104.242 255.255.255.255 192.168.51.4 255.255.255.255 tcp gt 1023 eq 23 non-secure route inbound l=y f=y t=3
permit 192.168.51.4 255.255.255.255 9.24.104.242 255.255.255.255 tcp/ack eq 23 gt 1023 non-secure route outbound l=y f=y t=3
# Service : Telnet in 2/2
# Description : Permit telnet in from firewall to secure network
permit 9.24.104.242 255.255.255.255 192.168.51.4 255.255.255.255 tcp gt 1023 eq 23 secure route outbound l=y f=y
permit 192.168.51.4 255.255.255.255 9.24.104.242 255.255.255.255 tcp/ack eq 23 gt 1023 secure route inbound l=y f=y
# Between NonSecureInt150.53.104.57 and RemFWNonSecInt
# Service : VPN encapsulation
# Description : Permit encrypted data between firewalls
permit 150.53.104.57 255.255.255.255 150.53.104.56 255.255.255.255 ah any 0 any 0 non-secure local both l=y f=y
permit 150.53.104.56 255.255.255.255 150.53.104.57 255.255.255.255 ah any 0 any 0 non-secure local both l=y f=y
permit 150.53.104.57 255.255.255.255 150.53.104.56 255.255.255.255 esp any 0 any 0 non-secure local both l=y f=y
permit 150.53.104.56 255.255.255.255 150.53.104.57 255.255.255.255 esp any 0 any 0 non-secure local both l=y f=y

```

Figure 380. Tunnel Filter Rules on Firewall 2 (FW2)

15.5.1.1 Log Files

Figure 381 shows us the log file on the sending firewall and how the packets are transported through the tunnel to the other tunnel partner and how the reply packet comes in.

Note: You will only see log entries if you use manual filters; dynamic filters do not log.

```

#:#13 R:p i:9.24.104.57 s:9.24.104.242 d:192.168.51.4 p:tcp sp:2540 dp:21 r:r a:s f:n T:0 e:n l:40
#:#19 R:p o:150.53.104.56 s:9.24.104.242 d:192.168.51.4 p:tcp sp:2540 dp:21 r:r a:n f:n T:4 e:n l:40
#:#31 R:p o:150.53.104.56 s:150.53.104.56 d:150.53.104.57 p:esp -:0 -:0 r:l a:n f:n T:0 e:n l:96
#:#32 R:p i:150.53.104.56 s:150.53.104.57 d:150.53.104.56 p:esp -:0 -:0 r:l a:n f:n T:0 e:n l:144
#:#20 R:p i:150.53.104.56 s:192.168.51.4 d:9.24.104.242 p:tcp sp:21 dp:2540 r:r a:n f:n T:4 e:n l:87
#:#14 R:p o:9.24.104.57 s:192.168.51.4 d:9.24.104.242 p:tcp sp:21 dp:2540 r:r a:s f:n T:0 e:n l:87
#:#25 R:p i:9.24.104.57 s:9.24.104.242 d:192.168.51.4 p:tcp sp:2541 dp:23 r:r a:s f:n T:0 e:n l:44
#:#27 R:p o:150.53.104.56 s:9.24.104.242 d:192.168.51.4 p:tcp sp:2541 dp:23 r:r a:n f:n T:3 e:n l:44
#:#29 R:p o:150.53.104.56 s:150.53.104.56 d:150.53.104.57 p:ah -:0 -:0 r:l a:n f:n T:0 e:n l:120
#:#30 R:p i:150.53.104.56 s:150.53.104.57 d:150.53.104.56 p:ah -:0 -:0 r:l a:n f:n T:0 e:n l:120
#:#28 R:p i:150.53.104.56 s:192.168.51.4 d:9.24.104.242 p:tcp sp:23 dp:2541 r:r a:n f:n T:3 e:n l:44
#:#26 R:p o:9.24.104.57 s:192.168.51.4 d:9.24.104.242 p:tcp sp:23 dp:2541 r:r a:s f:n T:0 e:n l:44

```

Figure 381. Traffic Logging In Sending Firewall

We can see that the FTP packet comes in at the secure interface (entry #:#13). From the nonsecure interface it is sent to Tunnel 4 (#:#19), encapsulated with new source and destination addresses from the tunnel endpoints and sent encrypted, (protocol:ESP) to the partner tunnel (#:#31 p:ESP). The reply packet comes back also encrypted (p:ESP in #:#32) and comes from Tunnel 4 (#:#20), where it will be unencrypted and decapsulated, and the secure interface sends it to its destination (#:#14). The next six lines in the log files are the same sequence for the Telnet traffic, now using Tunnel 3 and the AH protocol.

Figure 382 on page 387 shows us the log file on the receiving firewall and how the packets are seen in the partner tunnel endpoint.

```

#:28 R:p i:150.53.104.57 s:150.53.104.56 d:150.53.104.57 p:esp --:0 --:0 r:l a:n f:n T:0 e:n l:96
#:9 R:p i:150.53.104.57 s:9.24.104.242 d:192.168.51.4 p:tcp sp:2540 dp:21 r:r a:n f:n T:4 e:n l:40
#:15 R:p o:192.168.51.3 s:9.24.104.242 d:192.168.51.4 p:tcp sp:2540 dp:21 r:r a:s f:n T:0 e:n l:40
#:16 R:p i:192.168.51.3 s:192.168.51.4 d:9.24.104.242 p:tcp sp:21 dp:2540 r:r a:s f:n T:0 e:n l:87
#:10 R:p o:150.53.104.57 s:192.168.51.4 d:9.24.104.242 p:tcp sp:21 dp:2540 r:r a:n f:n T:4 e:n l:87
#:27 R:p o:150.53.104.57 s:150.53.104.57 d:150.53.104.56 p:esp --:0 --:0 r:l a:n f:n T:0 e:n l:144
#:28 R:p i:150.53.104.57 s:150.53.104.56 d:150.53.104.57 p:esp --:0 --:0 r:l a:n f:n T:0 e:n l:96
#:21 R:p i:150.53.104.57 s:9.24.104.242 d:192.168.51.4 p:tcp sp:2541 dp:23 r:r a:n f:n T:3 e:n l:44
#:23 R:p o:192.168.51.3 s:9.24.104.242 d:192.168.51.4 p:tcp sp:2541 dp:23 r:r a:s f:n T:0 e:n l:44
#:24 R:p i:192.168.51.3 s:192.168.51.4 d:9.24.104.242 p:tcp sp:23 dp:2541 r:r a:s f:n T:0 e:n l:44
#:22 R:p o:150.53.104.57 s:192.168.51.4 d:9.24.104.242 p:tcp sp:23 dp:2541 r:r a:n f:n T:3 e:n l:44
#:25 R:p o:150.53.104.57 s:150.53.104.57 d:150.53.104.56 p:ah --:0 --:0 r:l a:n f:n T:0 e:n l:120

```

Figure 382. Traffic Logging In Receiving Firewall

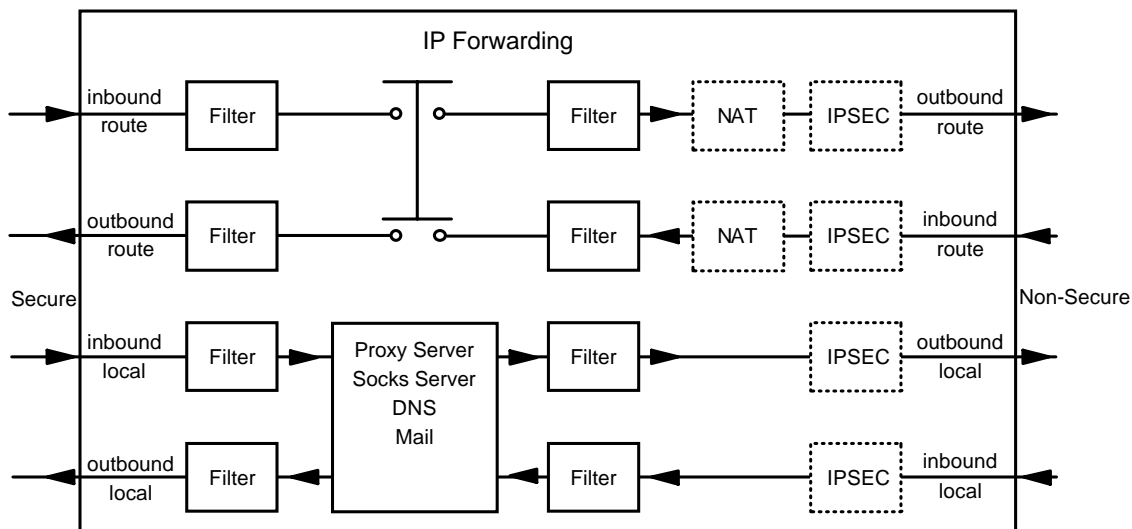
The first three lines show us how the ESP packet comes in from tunnel 4, is decrypted, decapsulated and send to his destination. The following three lines are the reply; the packet will be encapsulated and encrypted again, and sent to the other tunnel endpoint. The next six lines is the incoming traffic for the Telnet traffic and it follows the same flow as the FTP traffic.

15.5.1.2 Using NAT with IPSEC

NAT's purpose is to shield the IP addresses on the secure side of the firewall from the nonsecure side (see Chapter 14, "Network Address Translation" on page 339). This solves two problems:

- It allows you to use unregistered addresses in your secure network and still access the nonsecure network without conflict.
- It also keeps the nonsecure network hosts from knowing about any of your secure side host IP addresses.

To do this, NAT has to alter the source IP address of outgoing packets. When response packets come inbound on the connection, NAT reverses the translation it performed when the packet was outbound and resets the proper secure host's IP address in the packet destination fields. NAT does this even for packets that travel through tunnels as we see in Figure 383.

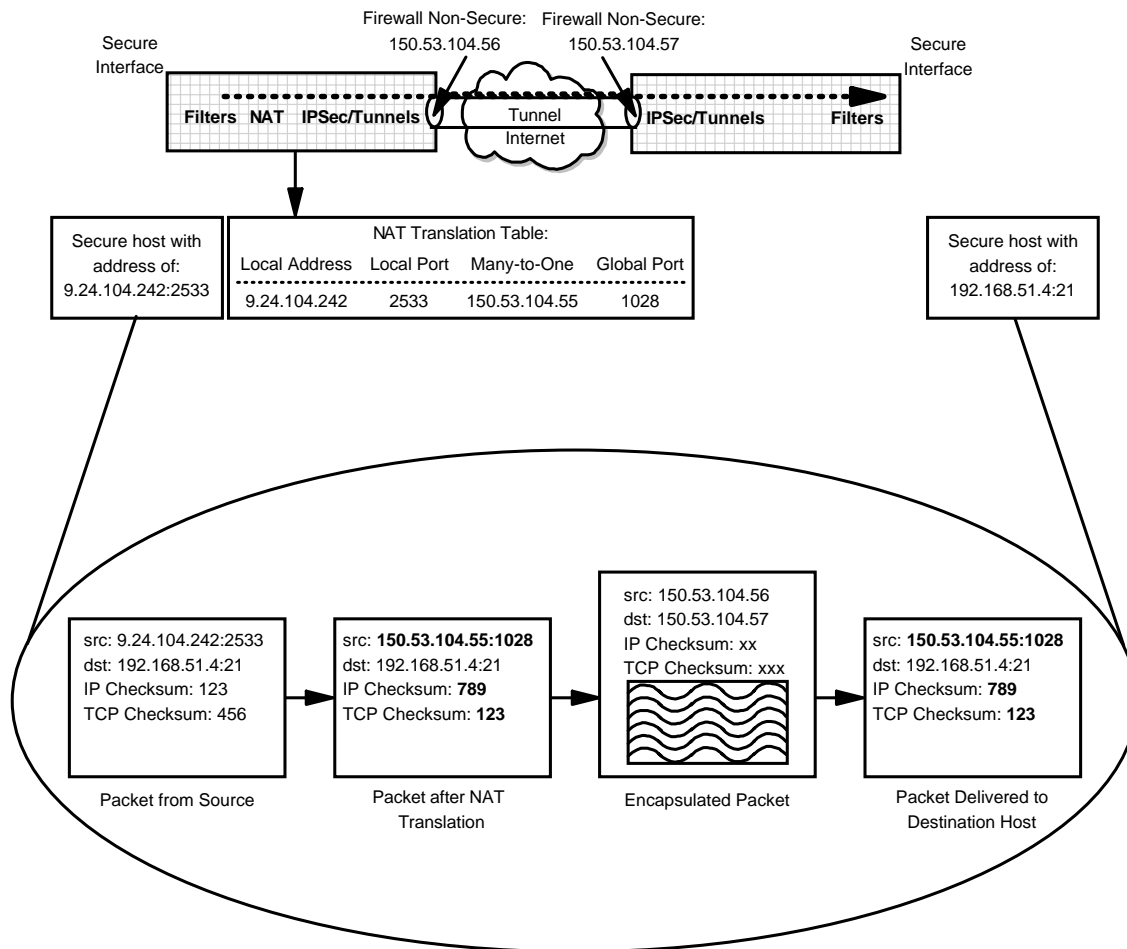


5209\FVPN048

Figure 383. Filters, NAT and IPSEC in the IBM eNetwork Firewall for Windows NT

Note:

- When we use the proxy server together with VPN, the first connection from the secure network to the secure interface of the firewall will be the ftp proxy 1/2. The tunnel data traffic connection will be from the nonsecure interface of the firewall to the remote secure network or host.
- We can create tunnels in two different ways: with dynamic or with static filter rules. When using NAT on the owning firewall we *cannot* simply use the dynamic filter rules. The reason is that dynamic filter rules are created automatically using the secure IP addresses for the traffic rules (see Figure 362 on page 371). When exporting and importing the tunnel definitions on the partner tunnel, the source and destination addresses will be reversed, but for the partner end the remote secure address is unknown because of NAT. On the partner end we *must* use the NAT address in the filter rule instead of the automatically created filter rule with the remote secure address. We could edit the export file and change the filter rules, but we think is safer to use static filter rules.



5209/FVPN047

Figure 384. Interaction between NAT, Filters and Tunnels

Figure 384 shows us the basic NAT translation. The bold fields in the second packet from the left illustrate the fields in the packet that are modified during outbound address translation. In general, filtering is applied to outbound packets prior to NAT and to inbound packets after NAT translation. Therefore, the filter rules are based on untranslated addresses. When NAT and tunnels are involved,

the filter rules at the firewall that has NAT active are also based on untranslated addresses. At the partner's end of the tunnel (assuming that NAT is not active at this firewall), the filter rules for inbound packets are based on translated source and destination addresses (for the inbound and outbound cases respectively). If NAT is active at both ends of the tunnel the discussion above applies in both directions.

We are still using the configuration as shown in Figure 374 on page 381 and we have only activated NAT on FW1 with NAT address 150.53.104.55. As a result the secure network 9.24.104.0 is no longer known to the outside world. Therefore we need to modify the filter rules on FW2, and use the NAT address 150.53.104.55 instead of the remote client's secure IP address 9.24.104.242.

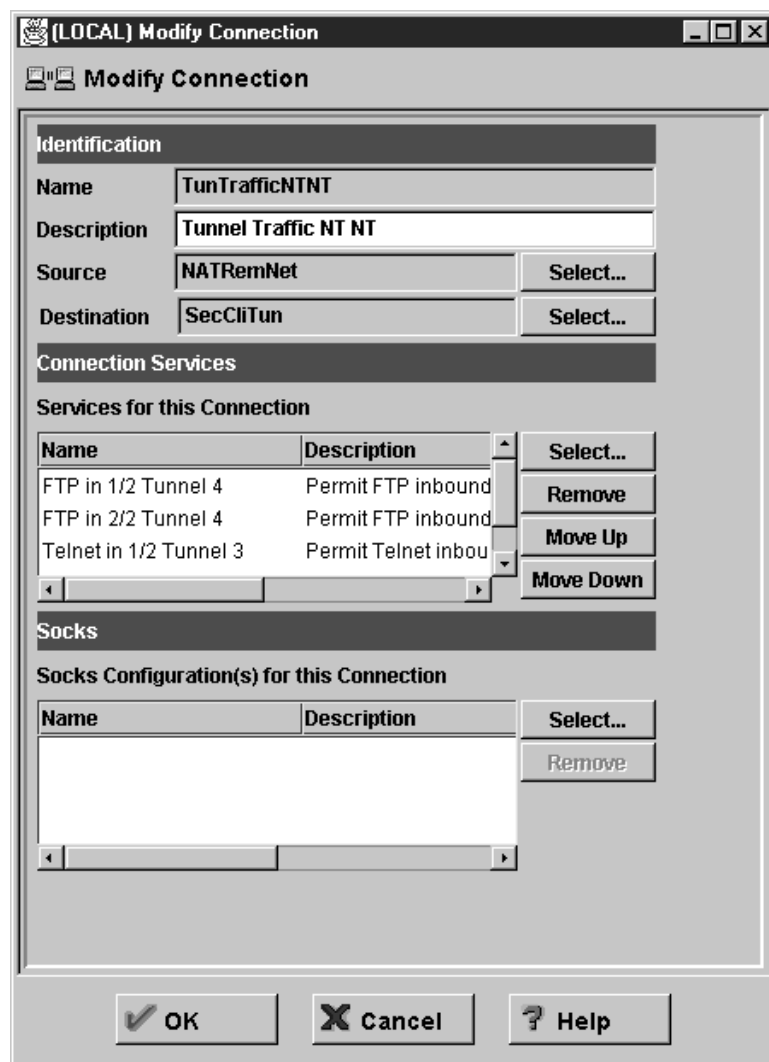


Figure 385. Traffic Connection Using NAT

In Figure 385 we see that the Source is the object with the remote NAT IP address.

```

# Between NatRemNet and SecCliTun
# Service : FTP in 1/2 Tunnel 4
# Description : Permit FTP inbound from non-secure network to firewall
permit 150.53.104.55 255.255.255.255 192.168.51.4 255.255.255.255 tcp gt 1023 eq 21 non-secure route inbound l=y f=y t=4
permit 192.168.51.4 255.255.255.255 150.53.104.55 255.255.255.255 tcp/ack eq 21 gt 1023 non-secure route outbound l=y f=y t=4
permit 192.168.51.4 255.255.255.255 150.53.104.55 255.255.255.255 tcp eq 20 gt 1023 non-secure route outbound l=y f=y t=4
permit 150.53.104.55 255.255.255.255 192.168.51.4 255.255.255.255 tcp/ack gt 1023 eq 20 non-secure route inbound l=y f=y t=4
permit 150.53.104.55 255.255.255.255 192.168.51.4 255.255.255.255 tcp gt 1023 gt 1023 non-secure route inbound l=y f=y t=4
permit 192.168.51.4 255.255.255.255 150.53.104.55 255.255.255.255 tcp/ack gt 1023 gt 1023 non-secure route outbound l=y f=y t=4
# Service : FTP in 2/2 Tunnel 4
# Description : Permit FTP inbound from firewall to secure network
permit 150.53.104.55 255.255.255.255 192.168.51.4 255.255.255.255 tcp gt 1023 eq 21 secure route outbound l=y f=y
permit 192.168.51.4 255.255.255.255 150.53.104.55 255.255.255.255 tcp/ack eq 21 gt 1023 secure route inbound l=y f=y
permit 192.168.51.4 255.255.255.255 150.53.104.55 255.255.255.255 tcp eq 20 gt 1023 secure route inbound l=y f=y
permit 150.53.104.55 255.255.255.255 192.168.51.4 255.255.255.255 tcp/ack gt 1023 eq 20 secure route outbound l=y f=y
permit 150.53.104.55 255.255.255.255 192.168.51.4 255.255.255.255 tcp gt 1023 gt 1023 secure route outbound l=y f=y
permit 192.168.51.4 255.255.255.255 150.53.104.55 255.255.255.255 tcp/ack gt 1023 gt 1023 secure route inbound l=y f=y
# Service : Telnet in 1/2
# Description : Permit Telnet inbound from non-secure network to firewall
permit 150.53.104.55 255.255.255.255 192.168.51.4 255.255.255.255 tcp gt 1023 eq 23 non-secure route inbound l=y f=y t=3
permit 192.168.51.4 255.255.255.255 150.53.104.55 255.255.255.255 tcp/ack eq 23 gt 1023 non-secure route outbound l=y f=y t=3
# Service : Telnet in 2/2
# Description : Permit telnet in from firewall to secure network
permit 150.53.104.55 255.255.255.255 192.168.51.4 255.255.255.255 tcp gt 1023 eq 23 secure route outbound l=y f=y
permit 192.168.51.4 255.255.255.255 150.53.104.55 255.255.255.255 tcp/ack eq 23 gt 1023 secure route inbound l=y f=y
# Between NonSecureInt150.53.104.57 and RemFWNonSecInt
# Service : VPN encapsulation
# Description : Permit encrypted data between firewalls
permit 150.53.104.57 255.255.255.255 150.53.104.56 255.255.255.255 ah any 0 any 0 non-secure local both l=y f=y
permit 150.53.104.56 255.255.255.255 150.53.104.57 255.255.255.255 ah any 0 any 0 non-secure local both l=y f=y
permit 150.53.104.57 255.255.255.255 150.53.104.56 255.255.255.255 esp any 0 any 0 non-secure local both l=y f=y
permit 150.53.104.56 255.255.255.255 150.53.104.57 255.255.255.255 esp any 0 any 0 non-secure local both l=y f=y

```

Figure 386. Filter Rules on Firewall 2 Using NAT on Firewall 1

Figure 386 shows us the new filter rules. The traffic rules for FTP and Telnet now contain the NAT address 150.53.104.55.

```

ICA1036i: #:13 R:p i:9.24.104.57 s:9.24.104.242 d:192.168.51.4 p:tcp sp:2533 dp:21 r:r a:s f:n T:0 e:n 1:44
ICA1036i: #:19 R:p o:150.53.104.56 s:9.24.104.242 d:192.168.51.4 p:tcp sp:2533 dp:21 r:r a:n f:n T:4 e:n 1:44
ICA09045i: NAT allocated address:port 150.53.104.55:1028 for secured address:port 9.24.104.242:2533
ICA1036i: #:31 R:p o:150.53.104.56 s:150.53.104.56 d:150.53.104.57 p:esp -:0 -:0 r:l a:n f:n T:0 e:n 1:96
ICA1036i: #:32 R:p i:150.53.104.56 s:150.53.104.57 d:150.53.104.56 p:esp -:0 -:0 r:l a:n f:n T:0 e:n 1:96
ICA1036i: #:20 R:p i:150.53.104.56 s:192.168.51.4 d:9.24.104.242 p:tcp sp:21 dp:2533 r:r a:n f:n T:4 e:n 1:44
ICA1036i: #:14 R:p o:9.24.104.57 s:192.168.51.4 d:9.24.104.242 p:tcp sp:21 dp:2533 r:r a:s f:n T:0 e:n 1:44

```

Figure 387. NAT Log File Sending Firewall

In the log file (see Figure 387), we see the packet flow on the sending firewall. The packet comes in and the filter rules are applied, rules #:13 and #:19. Then NAT interacts and changes the source to the NAT address. The next rule sends the packet to the other side of the tunnel using the ESP protocol (rule #:31). Rule #:32 receives the reply, decrypts the packet and sends it to port 1028 for NAT translation. Then the filter rules are applied and the packet is sent to its destination, rules #:20 and #:14.

```

ICA1036i: #:28 R:p i:150.53.104.57 s:150.53.104.56 d:150.53.104.57 p:esp -:0 -:0 r:l a:n f:n T:0 e:n 1:96
ICA1036i: #:9 R:p i:150.53.104.57 s:150.53.104.55 d:192.168.51.4 p:tcp sp:1028 dp:21 r:r a:n f:n T:4 e:n 1:44
ICA1036i: #:15 R:p o:192.168.51.3 s:150.53.104.55 d:192.168.51.4 p:tcp sp:1028 dp:21 r:r a:s f:n T:0 e:n 1:44
ICA1036i: #:16 R:p i:192.168.51.3 s:192.168.51.4 d:150.53.104.55 p:tcp sp:21 dp:1028 r:r a:s f:n T:0 e:n 1:44
ICA1036i: #:10 R:p o:150.53.104.57 s:192.168.51.4 d:150.53.104.55 p:tcp sp:21 dp:1028 r:r a:n f:n T:4 e:n 1:44
ICA1036i: #:27 R:p o:150.53.104.57 s:150.53.104.57 d:150.53.104.56 p:esp -:0 -:0 r:l a:n f:n T:0 e:n 1:96

```

Figure 388. NAT Log File Receiving Firewall

At the receiving side of the tunnel (see Figure 388), the packet that comes in using ESP protocol, rule #:28; the IP addresses are the nonsecure adapters of both firewalls. After decapsulation, we see that the source address and source port are respectively the NAT address and the global port added by NAT, rule #:9. The packet is sent to its destination, rule #:15, and the reply comes back, rule #:16.

The destination of the packet is the NAT address and port, and will be sent back using the ESP protocol, rules #:10 and #:20.

15.5.2 Virtual Private Network between IBM eNetwork Firewall for Windows NT and AIX V4.3

The next scenario is a tunnel between the IBM eNetwork Firewall for Windows NT and the AIX 4.3.2 operating system. We have a secure network (192.168.100.0) with an NT 4.0 client (192.168.100.7) and a nonsecure network (172.16.3.0) with the AIX connected. The tunnel is configured between the NT firewall with nonsecure interface (172.16.3.5) and the AIX with the nonsecure interface (172.16.3.1). The AIX is also the client in this configuration. Figure 389 shows us the configuration.

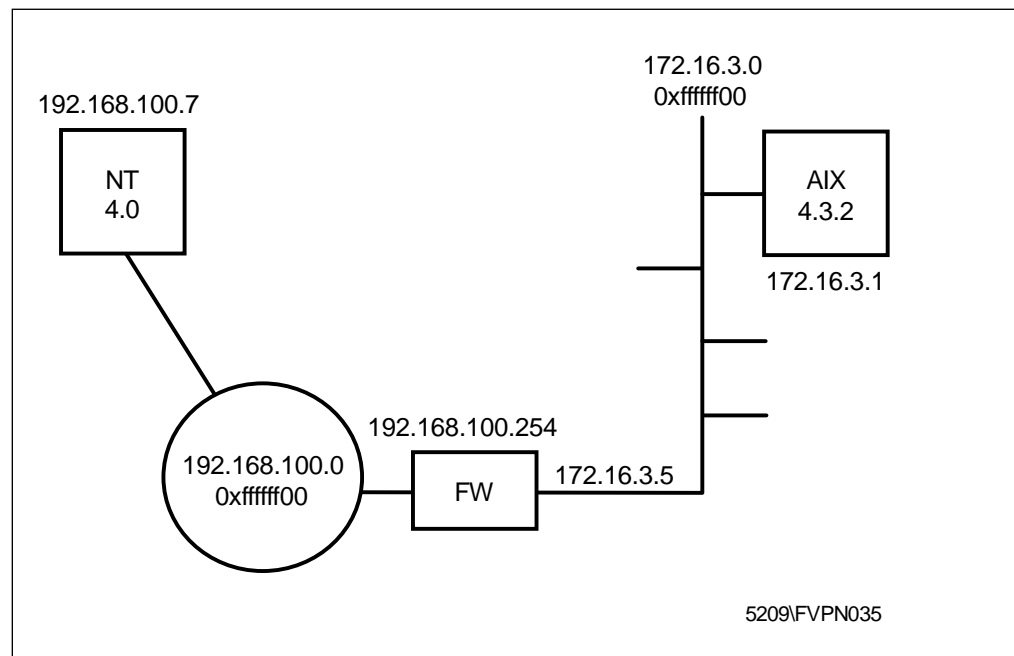


Figure 389. Network Tunnel Configuration between IBM eNetwork Firewall for Windows NT and AIX 4.3

We executed the following steps:

- On the NT firewall we have created a tunnel with dynamic filter rules. This means the filter rules are put in memory when we activate the tunnel and activated when we activate the tunnel (see Figure 390 on page 392). You can see the dynamic filter rules in Figure 362 on page 371.

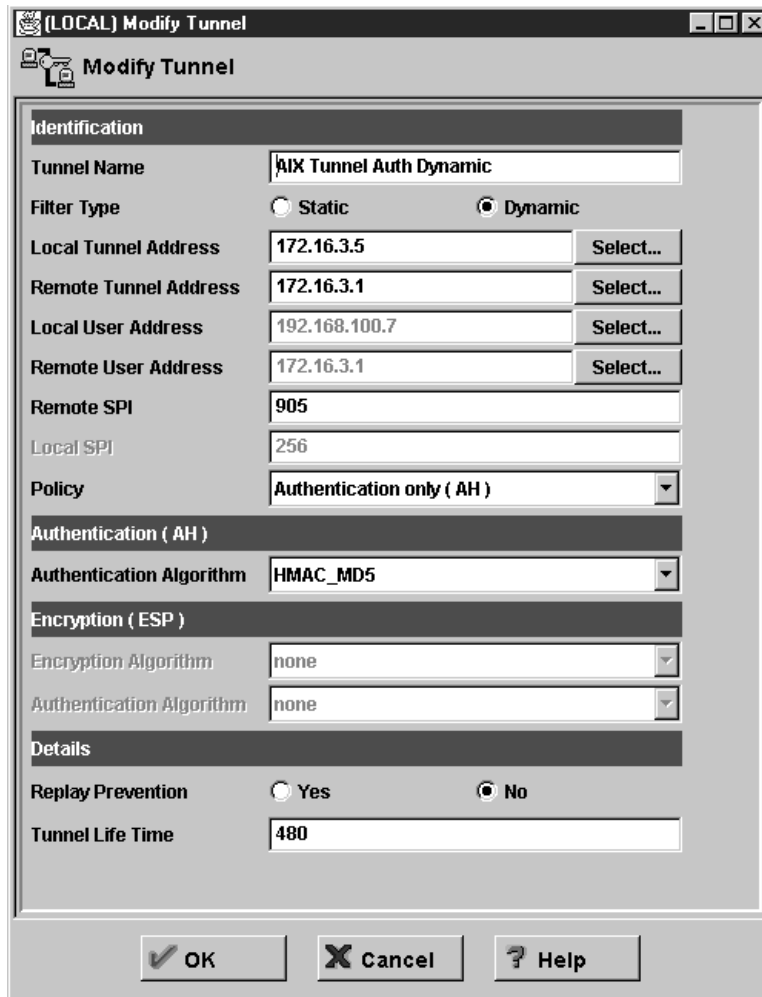


Figure 390. Tunnel Definition between IBM eNetwork Firewall for Windows NT and AIX 4.3

- We export the tunnel so the ipsec_tun_man.exp file is created.

```
#-----  
4  
172.16.3.5  
172.16.3.1  
1  
905  
905  
256  
256  
NONE  
0  
0x  
NONE  
0  
0x  
HMAC_MD5  
16  
0xb14dfbc1fc4bd1a3f633908a8c1cfa2f  
HMAC_MD5  
16  
0xd5b0003caba5ca5697a7c6e25c527f91  
0  
28800  
tunnel  
tunnel  
axax  
0  
1  
NONE  
0  
  
NONE  
0  
  
0  
-  
-  
AIX Tunnel Auth Dynamic  
1  
192.168.100.7  
255.255.255.255  
172.16.3.1  
255.255.255.255
```

Figure 391. Export File

- Before importing the file on the AIX we need to convert the file format of the export file to an AIX file format. The `conv_export_file` utility will execute the conversion for us. The command syntax is:

```
conv_export_file dir=dddd
```

where `dir=dddd` specifies the directory of the location of the export file to be converted, as shown in Figure 392.

```
C:\Program Files\IBM\Firewall\bin>conv_export_file dir=C:\vpn_nt_aix  
Command completed successfully.  
  
C:\Program Files\IBM\Firewall\bin>
```

Figure 392. Successfully Converted Export File

Note: The conversion utility comes with the IBM eNetwork Firewall for Windows NT and runs on Windows NT. The `conv_export_file` utility can only be used for a tunnel between the IBM eNetwork Firewall for Windows NT and AIX. The converter utility will modify the existing file; it does not create a new file.

```
#-----  
4  
172.16.3.5  
172.16.3.1  
1  
905  
905  
256  
256  
NONE  
0  
0x  
NONE  
0  
0x  
HMAC_MD5  
16  
0xb14dfbc1fc4bd1a3f633908a8c1cfa2f  
HMAC_MD5  
16  
0xd5b0003caba5ca5697a7c6e25c527f91  
0  
28800  
tunnel  
tunnel  
axax  
0  
1  
NONE  
0  
  
NONE  
0  
  
0  
-  
-
```

Figure 393. Converted Export File

The only difference we noticed between the original file and the converted file is the deletion of the filter lines at the bottom of the file.

When the export file is successfully converted we can import the tunnel on the AIX server. To import follow the next steps:

1. Execute the DOSREAD Command: `dosread -a IPSEC_1.EXP ipsec_tun_manu.exp`

Note: Be sure the DOS utilities are installed on AIX

2. Start IP Security
3. Go to the Basic IP Security configuration menu and import the export file
4. Modify the filter rules (see Figure 394 on page 395). Add the traffic rule between the two clients
5. Activate the rules
6. Activate the tunnel

```

1 *** Dynamic filter placement rule for IKE tunnels * * * no
2 permit 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 yes ah any 0 any 0 both both no all packets 0 all
3 permit 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 yes esp any 0 any 0 both both no all packets 0 all
4 permit 172.16.3.5 255.255.255.255 172.16.3.1 255.255.255.255 yes all any 0 any 0 both inbound yes all packets 1 all
5 permit 172.16.3.1 255.255.255.255 172.16.3.5 255.255.255.255 yes all any 0 any 0 both outbound yes all packets 1 all
6 permit 172.16.3.1 255.255.255.255 192.168.100.7 255.255.255.255 yes all any 0 any 0 local outbound no all packets 1 all
7 permit 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 yes all any 0 any 0 both both no all packets 0 all

```

Figure 394. AIX Tunnel Rules

AIX rule description:

The AIX filter rules syntax is a little different from those in the firewall: there are no secure or nonsecure interfaces.

1. Title 1 and rule 7: Two default rules automatically created by the system. Note that rule 7 opens the system to any traffic; this is not a firewall, so there is no need to restrict the traffic, but you may change it if you want.
2. Rules 2 and 3: Automatically created when the IP Security is started on the system.
3. Rules 4 and 5: Automatically created when importing the export file. The number 1 refers to the tunnel ID.
4. Rule 6: This rule we *must* create manually. This rule defines the traffic between the two clients. The number 1 refers to the tunnel ID.

15.5.3 Virtual Private Network between IBM eNetwork Firewall for Windows NT and OS/390

We assume we will configure a tunnel between the IBM eNetwork Firewall for Windows NT and the beta version of OS/390 Firewall Technologies V2R8 using the default static filter rules. In Figure 395 we see the configuration; the traffic is configured between the secure client 192.168.10.6 and the OS/390.

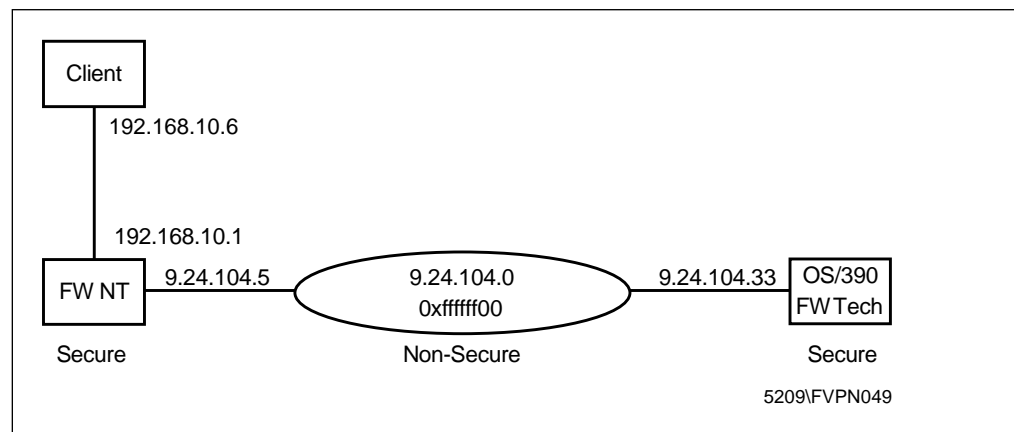


Figure 395. Network Tunnel Configuration between IBM eNetwork Firewall for Windows NT and OS/390 V2R8 beta

We define a tunnel with only authentication, using algorithm HMAC_MD5; do not use KEYED_MD5 because it is not supported by IBM eNetwork Firewall for Windows NT (not supported in the newer RFCs). We use the GUI to define the tunnel, as shown in Figure 396 on page 396.

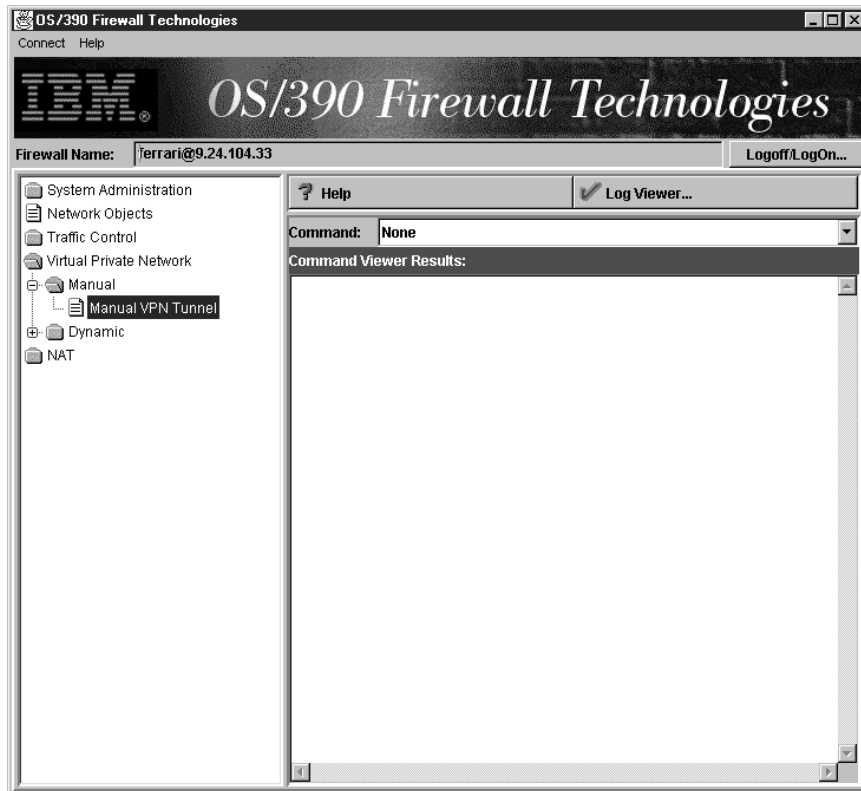


Figure 396. GUI in OS/390 V2R8 beta

Figure 397 on page 397, Figure 398 on page 397 and Figure 399 on page 398 show the screens we used to define the tunnel.

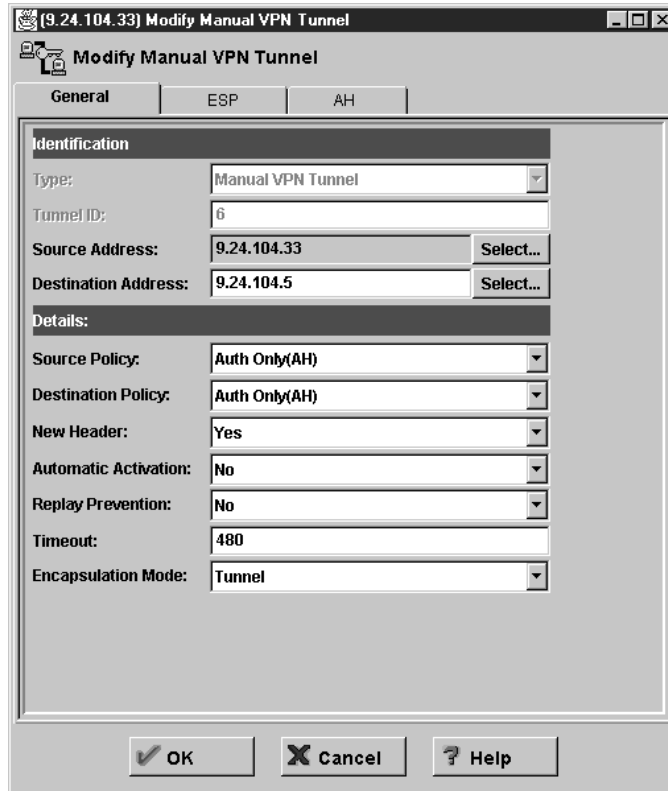


Figure 397. Tunnel Definition in OS/390 V2R8 beta (Part 1 of 3)

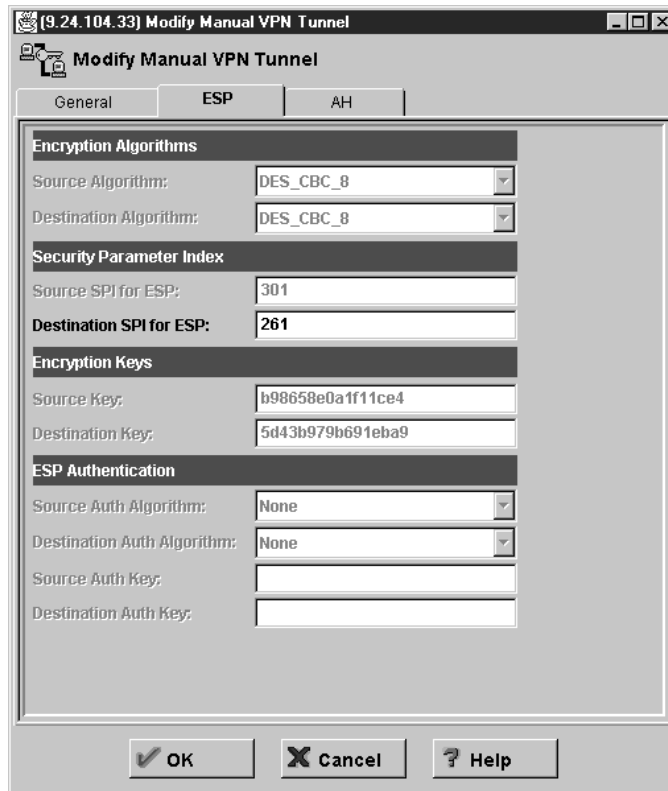


Figure 398. Tunnel Definition in OS/390 V2R8 beta (Part 2 of 3)

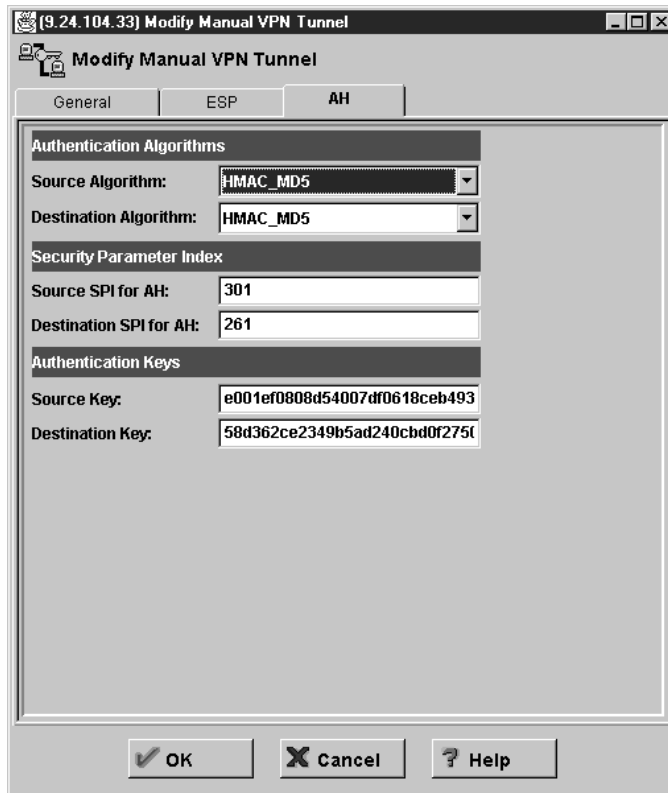


Figure 399. Tunnel Definition in OS/390 V2R8 beta (Part 3 of 3)

We then export the tunnel definition; OS/390 creates two files, as shown in Figure 400 on page 399 and Figure 401 on page 399.


```

#Tunnel_partner Version = 1.0
9.24.104.33
9.24.104.5
6
261
261
301
301
DES_CBC_8
8
0x5d43b979b691eba9
DES_CBC_8
8
0xb98658e0a1f11ce4
HMAC_MD5
16
0x58d362ce2349b5ad240cbd0f2750a853
HMAC_MD5
16
0xe001ef0808d54007df0618ceb4931258
0
28800
ESP_NOOP
0
0x
ESP_NOOP
0
0x
n
n
n
n
y
y
y
y

```

Figure 400. Export File fwexpmctx.manual Created in OS/390

A basic description of the fields of this file are in Table 33 on page 400. In our case we see keys for encryption, but they are not used. If we would use encryption, we should choose an encryption algorithm that is supported by both firewalls. OS/390 has two flavors for DES: DES_CBC_4 and DES_CBC_8; the latter is the one that matches the DES algorithm of IBM eNetwork Firewall for Windows NT.

```

9.24.104.5 9.24.104.33 6 28800 n/y n/n y key=manual n/y y

```

Figure 401. Export File fwexppolicy Created in OS/390

Table 33 (Page 1 of 2). Export File Format for OS/390 Version 2 Release 7

Typical Contents	File Format
#Tunnel_partner Version = 1.0	context line - must be #Tunnel_partner Version = 1.0
9.24.104.05	IP Address of Tunnel Owner
9.24.104.43	IP Address of Tunnel Partner
6	Tunnel ID
800	ESP SPI of Tunnel Partner
800	AH SPI of Tunnel Partner
260	ESP SPI of Tunnel Owner
260	AH SPI of Tunnel Owner
CDMF	Encryption Algorithm of Tunnel Partner (CDMF or DES_CBC_8 or DES_CBC_4 or 3DES_CBC or ESP_NULL). IBM eNetwork Firewall for Windows NT does not support DES_CBC_4.
8 for CDMF or DES_CBC; 16 for triple DES	Size (number of bytes) of encryption Key of Tunnel Partner
0x.. (8 Bytes in Hex)	Encryption Key of Tunnel Partner (hex value starting with '0x')
CDMF	Encryption Algorithm of Tunnel Owner (CDMF or DES_CBC_8 or DES_CBC_4 or 3DES_CBC or ESP_NULL). IBM eNetwork Firewall for Windows NT does not support DES_CBC_4.
8 for CDMF or DES_CBC; 16 for triple DES	Size (number of bytes) of encryption Key of Tunnel Owner
0x.. (8 Bytes in Hex)	Encryption Key of Tunnel Owner (hex value starting with '0x')
HMAC_MD5	Authentication Algorithm of Tunnel Partner (KEYED_MD5 or HMAC_MD5 or HMAC_SHA). IBM eNetwork Firewall for Windows NT does not support KEYED_MD5.
16	size (number of bytes) of authentication Key of Tunnel Partner
0x.. (16 Bytes in Hex)	Authentication Key of Tunnel Partner (hex value starting with '0x')
HMAC_MD5	Authentication Algorithm of Tunnel Owner (KEYED_MD5 or HMAC_MD5 or HMAC_SHA). IBM eNetwork Firewall for Windows NT does not support KEYED_MD5.
16	size (number of bytes) of Authentication Key of Tunnel Owner
0x.. (16 Bytes in Hex)	Authentication Key of Tunnel Owner (hex value starting with '0x')
0	Reserved set to 0
28800	Timeout in seconds
ESP_NOOP	ESP Authentication Algorithm of Tunnel Partner (HMAC_MD5 or HMAC_SHA or ESP_NOOP)
0	size (number of bytes) of ESP Authentication Key of Tunnel Partner
0x	ESP Authentication Key of Tunnel Partner (hex value starting with '0x')
ESP_NOOP	ESP Authentication Algorithm of Tunnel Owner (HMAC_MD5 or HMAC_SHA or ESP_NOOP)
0	size (number of bytes) of ESP Authentication Key of Tunnel Owner
0x	ESP Authentication Key of Tunnel Owner (hex value starting with '0x')

Table 33 (Page 2 of 2). Export File Format for OS/390 Version 2 Release 7

Typical Contents	File Format
n	ESP replay prevention indicator for tunnel partner (n - no replay, y - replay)
n	ESP replay prevention indicator for tunnel owner (n - no replay, y - replay)
n	AH replay prevention indicator for tunnel partner (n - no replay, y - replay)
n	AH replay prevention indicator for tunnel owner (n - no replay, y - replay)
n	ESP new header indicator for tunnel partner (n - no new header, y - new header)
n	ESP new header indicator for tunnel owner (n - no new header, y - new header)
n	AH new header indicator for tunnel partner (n - no new header, y - new header)
n	AH new header indicator for tunnel owner (n - no new header, y - new header)

To import the export file into the IBM eNetwork Firewall for Windows NT we *must* modify the file manually. We must modify fwexpmctx.manual to match the format described in Table 32 on page 366, as shown in Figure 402 on page 402

```

#-----
4
9.24.104.33
9.24.104.5
6
261
261
301
301
DES_CBC
8
0x5d43b979b691eba9
DES_CBC
8
0xb98658e0a1f11ce4
HMAC_MD5
16
0x58d362ce2349b5ad240cbd0f2750a853
HMAC_MD5
16
0xe001ef0808d54007df0618ceb4931258
0
28800
tunnel
tunnel
axax
0
1
NONE
0

NONE
0

0
-
-
DEF_IN_390
0

```

Figure 402. Modified File

After importing the file into IBM eNetwork Firewall for Windows NT, the tunnel looks like the following:

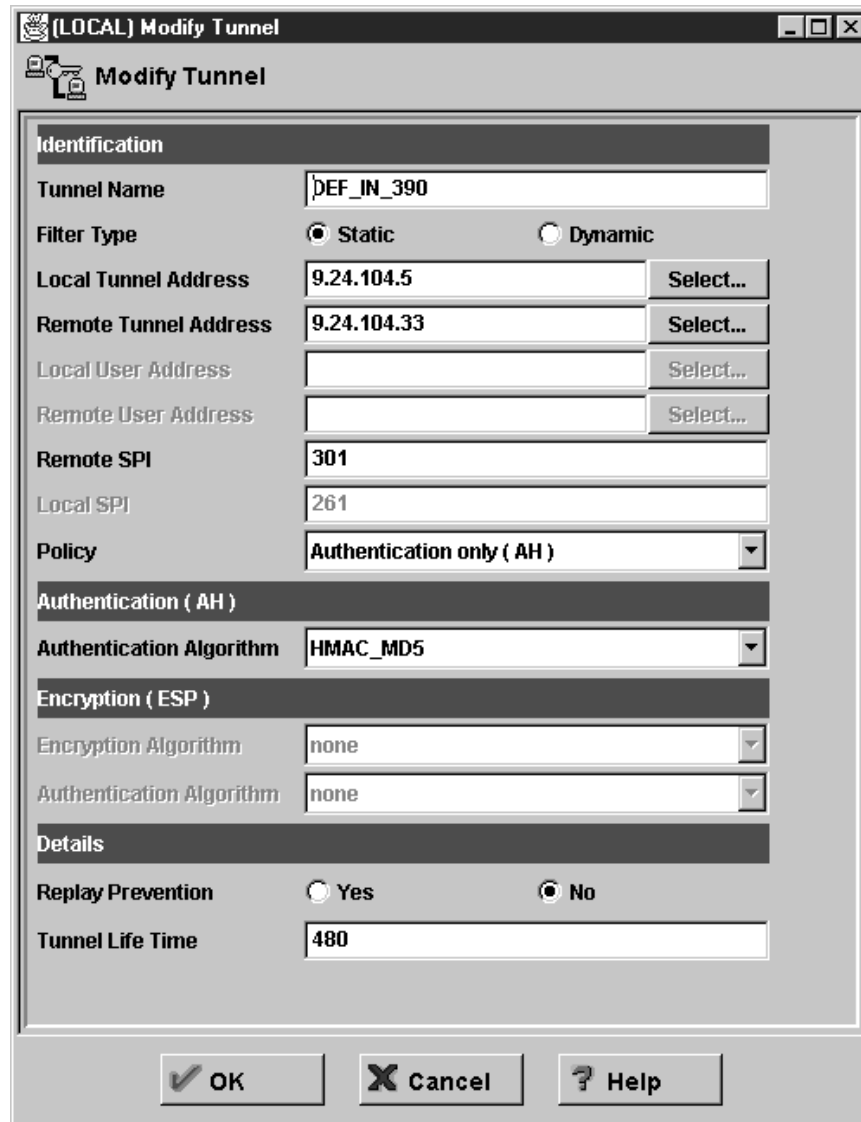


Figure 403. Tunnel Definition of Imported File

We must now create the filter rules in each firewall. The filter rules in OS/390 look like the following:

```

permit 9.24.104.33 255.255.255.255 9.24.104.5 255.255.255.255 ah any 0 any 0 secure local both l=y f=y s=m d=m;
permit 9.24.104.5 255.255.255.255 9.24.104.33 255.255.255.255 ah any 0 any 0 secure local both l=y f=y s=m d=m;
permit 9.24.104.33 255.255.255.255 192.168.10.0 255.255.255.0 all any 0 any 0 both both both l=y f=y t=6 s=m d=m;
permit 192.168.10.0 255.255.255.0 9.24.104.33 255.255.255.255 all any 0 any 0 both both both l=y f=y t=6 s=m d=m;

```

Figure 404. Filter Rules in OS/390

These rules need some clarifications.

- In the firewall for OS/390, 9.24.104.33 was the secure network, that's why the rules allows AH traffic through the secure interface.
- The endpoint of the tunnel in OS/390 was the same host where the firewall resides.
- The last two rules should refer to the secure interface rather than 'both'.

You should follow the rules we described earlier in this chapter to create safe filters.

In the IBM eNetwork Firewall for Windows NT, the filter rules were:

```
permit 192.168.10.6 255.255.255.255 9.24.104.33 255.255.255.255 all any 0 any 0 secure route inbound l=y f=y t=0 e=none a=none
permit 9.24.104.33 255.255.255.255 192.168.10.6 255.255.255.255 all any 0 any 0 secure route outbound l=y f=y t=0 e=none a=none
permit 192.168.10.6 255.255.255.255 9.24.104.33 255.255.255.255 all any 0 any 0 non-secure route outbound l=y f=y t=6 e=none a=none
permit 9.24.104.33 255.255.255.255 192.168.10.6 255.255.255.255 all any 0 any 0 non-secure route inbound l=y f=y t=6 e=none a=none
permit 9.24.104.5 255.255.255.255 9.24.104.33 255.255.255.255 ah any 0 any 0 non-secure local both l=y f=y t=0 e=none a=none
permit 9.24.104.33 255.255.255.255 9.24.104.5 255.255.255.255 ah any 0 any 0 non-secure local both l=y f=y t=0 e=none a=none
permit 9.24.104.5 255.255.255.255 9.24.104.33 255.255.255.255 esp any 0 any 0 non-secure local both l=y f=y t=0 e=none a=none
permit 9.24.104.33 255.255.255.255 9.24.104.5 255.255.255.255 esp any 0 any 0 non-secure local both l=y f=y t=0 e=none a=none
```

Figure 405. Filter Rules in IBM eNetwork Firewall for Windows NT

After activating the tunnels, we FTP from 192.168.10.6 to 9.24.104.33. The IBM eNetwork Firewall for Windows NT and the OS/390 logs showed that the traffic was flowing through tunnel ID=6.

15.5.4 VPN between IBM eNetwork Firewall for Windows NT and OS/400 V4R4

In the next scenario we configured a tunnel with static filter rules between the IBM eNetwork Firewall for Windows NT and the AS/400 operating system R4V4. Figure 406 shows us the network configuration.

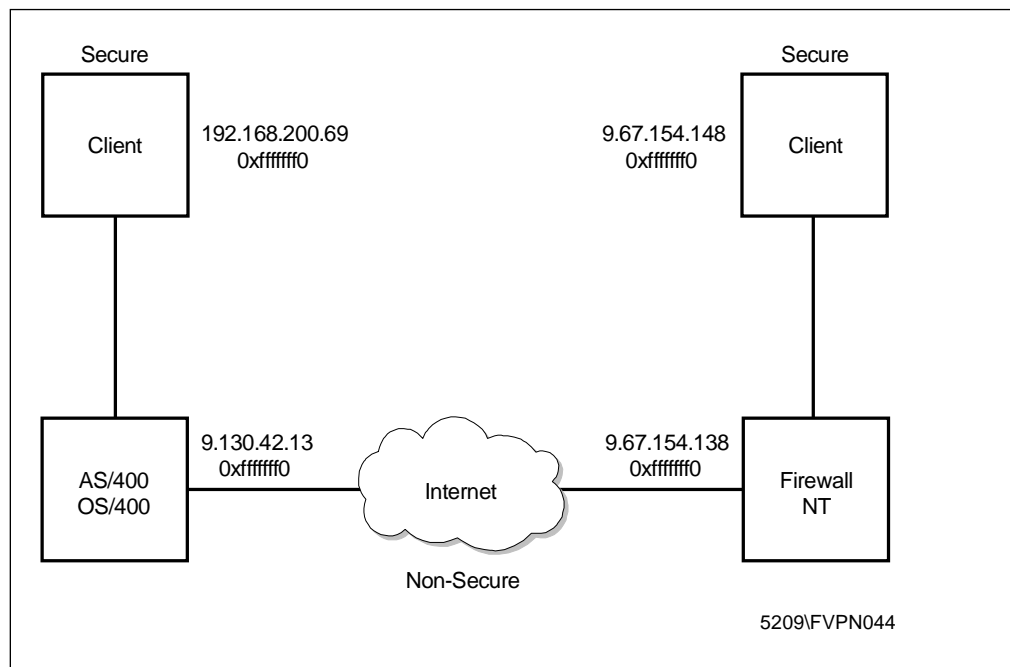


Figure 406. Network Tunnel Configuration between IBM eNetwork Firewall for Windows NT and AS/400 V4R4

When configuring a tunnel with the AS/400 operating system we cannot use authentication and encryption together, we *must* choose one.

We configured the tunnel on the IBM eNetwork Firewall for Windows NT. The tunnel on the AS/400 system must be created manually because the AS/400 system does not have an import function.

To do this we must follow the next steps:

- Open the AS/400 Operations Navigator window.
- Select **Network**.
- Select **IP Security**. On the right side we can select IP Packet Security to define the filter rules, and Virtual Private Network to configure the tunnel definitions.

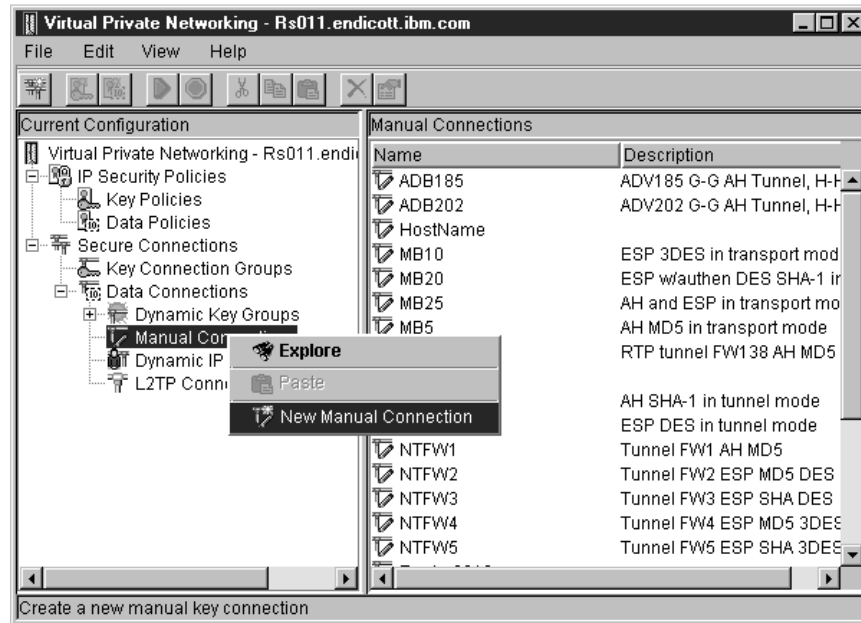


Figure 407. AS/400 Virtual Private Networking Window

We must create the tunnel first by following these steps:

- Select **Virtual Private Network**
- Select **Secure Connections**
- Select **Data Connections**
- Right-click **Manual Connections**
- Select **New Manual Connection**

All these steps are shown in Figure 407. After selecting the new manual connection we get the Properties panel, as shown in Figure 408 on page 406, where we configure the connection.

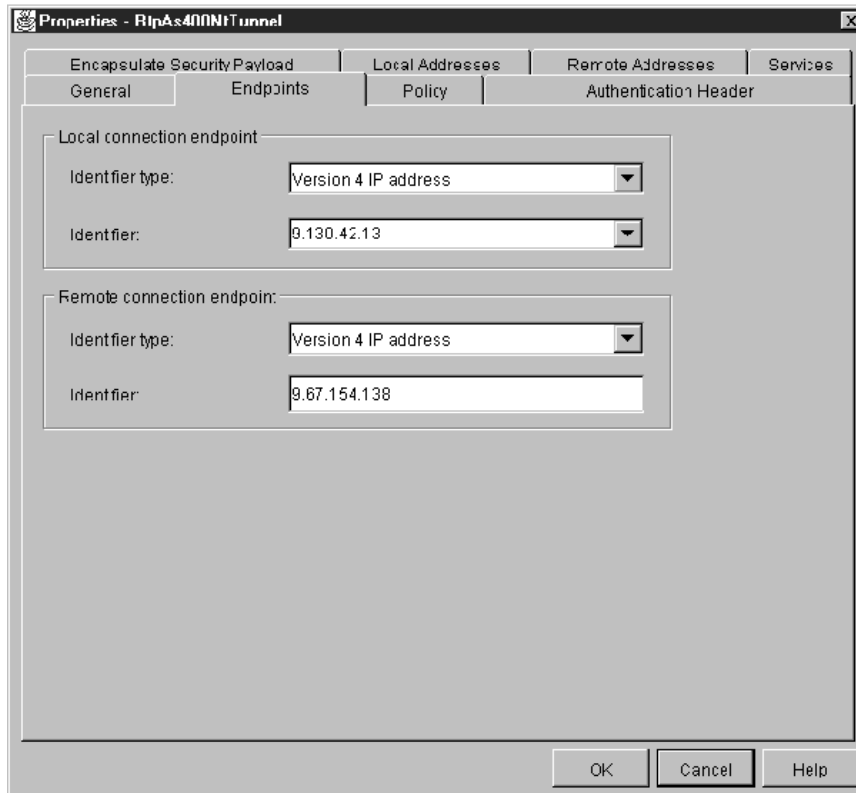


Figure 408. Add Manual Connection.

To configure the tunnel, follow the next steps:

1. Configure the connection

- **General:** Enter Name, Description and System Role. System Role defines if the firewall acts as a gateway or a host at both end points: local and remote.
- **Endpoints:** Enter the two endpoints of the tunnel, select the type, and enter the identifier. In our example the type is Version 4 IP Address and we entered the IP addresses (see Figure 408).
- **Policy:** Enter the policy. We have to select either authentication or encryption, not both and the tunnel lifetime.
- **Authentication Header:** Enter the authentication method and the inbound and outbound SPIs and Keys.

a. Inbound SPI = Remote SPI from the tunnel owner (Line 6).

b. Outbound SPI = Local SPI from the tunnel owner (Line 8).

Note: The SPIs on the AS/400 must be entered in HEX form and must have a length of 8. Example SPI 2000 will be 000007d0.

c. Inbound key = Enter the key from the export file (line 20).

d. Outbound key = Enter the key from the export file on (line 17).

Note: The inbound and outbound keys must be entered manually. Be careful when entering, one mistake and the tunnel will not work. For lines 6, 8 17 and 20, refer to Table 32 on page 366.

- Encapsulate Security Payload: Enter the inbound and outbound SPIs and keys the same way as described under authentication.
- Local Addresses: Enter the secure client or network address and the type; in our example, 192.168.200.69 and Version 4.
- Remote Addresses: Enter the remote secure client or network address and the type; in our example 9.67.154.148 and Version 4.
- Services: Enter the ports and protocols we want to allow to be transferred through the tunnel. We selected any local and remote ports and all protocols to let all traffic go through.

2. Configure the IP filter rules

After configuring the tunnel, we will configure the filter rules. We create the filter for the encapsulation, the IPSec filter, and the filters for the traffic on the secure side. When creating the filters we must enter a setname, a connection name, the clients addresses and the permitted protocols. The setname contains all the filters for one tunnel.

3. After creating the filters we create the filter interface, where we define what interface and what setname we will be using.
4. Activate the filters at both ends.
5. Activate the tunnel at both ends.

Note: If you request replay prevention in a manual tunnel and you de-activate the tunnel in the NT side, the re-activation will look like it failed because the flow of traffic will be delayed. After re-activation, the Windows NT firewall will start with replay number zero. The AS/400 firewall will discard tunnel traffic because it is expecting a number higher than the last received. The flow of traffic will resume after OS/400 receives the replay number it was expecting. This situation occurs not only with OS/400, but with any other partner that supports replay prevention.

Chapter 16. Logging, Monitoring, and Reporting

Logging is essential to the day-to-day operation of the IBM Firewall. Unless you log the activity on your firewall and generate alerts for suspicious activity, you could be under attack without even realizing it. Worse, in the event of an attack, you would be seriously hampered in your attempts to determine the origin and target of the attack.

This chapter describes how to configure the logging facilities, how to monitor the logging of alerts in real-time, and how to build useful reports on top of the logged data. The IBM Firewall monitors the messages sent to the firewall log for potential crisis situations, based upon user-defined thresholds. In the event of a threshold violation, the firewall delivers an alert, in a manner specified by the firewall administrator.

Configuration Files

We give you many configuration file names in this chapter. Don't worry if your file names begin with `C:\PROGRA~1\IBM\Firewall\` instead of with `C:\Program Files\IBM\Firewall\`. Both paths point to the same directory. The short version is used for applications which can handle only directory or file names up to eight characters.

16.1 Configure Logging

When the IBM Firewall is initially installed, the logging facilities are preconfigured for basic operation. However, you need to change this configuration to suit your environment.

16.1.1 Logging Priority Levels

Let's start with a quick look at *syslog* to understand the basics of logging. Syslog is a daemon that comes from the UNIX environment. It is used to centralize the logging activities from different applications. UNIX systems provide eight *priority levels* to specify the amount of logging activity for a specific service. They are:

- Debug
- Information
- Notice
- Warning
- Error
- Critical
- Alert
- Emergency

At the *debug* level all activity for that facility will be logged. At the *emergency* level, very little activity (only severe messages) would be logged at all.

On Windows NT the syslog service is done by the IBM Firewall Log Daemon. The parameter file for this service is `C:\Program Files\IBM\Firewall\config\syslog.conf`.

You should, however, use the Configuration Client to configure the log facilities as it will take archive settings into account and refresh the process for you.

The IBM Firewall only uses five of the available priority levels, which are shown in Table 34.

Table 34. IBM Firewall Log Priority Levels

Level	Description
debug	All messages are logged.
information	Only messages with the priority levels information, warning, error, and critical are logged.
warning	Only messages with the priority levels warning, error, and critical are logged.
error	Only messages with the priority levels error and critical are logged.
critical	Only messages with the priority level critical are logged.

16.1.2 Log Facilities

The log facility determines the type and source of information that is logged. You can select the log facilities shown in Table 35.

Table 35. IBM Firewall Log Facilities

Facility	Internal Name	Description
Firewall Log	local4	General firewall log, including IP filter logging, SOCKS and proxy usage, and mail events.
Alert Log	local1	Log monitor threshold violation warnings. The messages displayed in the Alert Display (see Figure 437 on page 430) are taken from this facility. You get messages in the Alert Display only if you create a alert log file.
Admin Audit Log	local0	All firewall administrator functions are logged here.

16.1.3 Manage Log Facilities

You can manage the first three log facilities from Table 35 with the Configuration Client. The Admin Audit Log facility can only be configured and browsed by the command line (see 16.1.5, "Manage the Admin Audit Log Facility" on page 417).

To create new log facilities or modify existing ones select **System Administration** → **System Logs** → **Log Facilities** from the Configuration Client navigation tree.

You should now see Figure 409 on page 411.

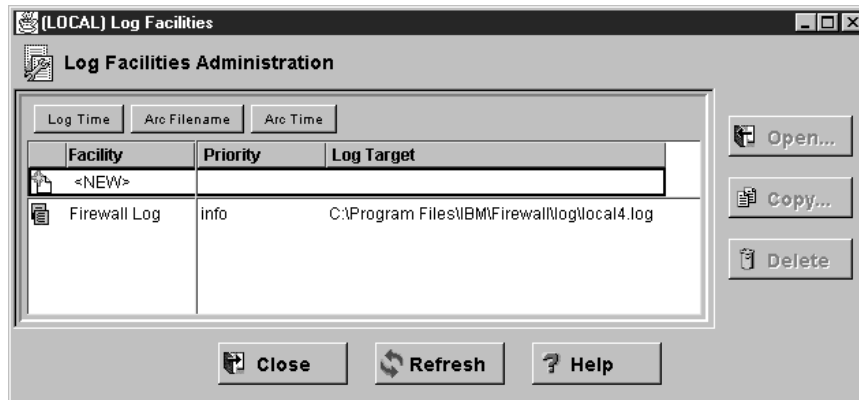


Figure 409. Log Facilities Administration

As you can see, the Firewall Log facility is already created. At this time it logs all entries that have at least the information level of priority.

To create a new log facility to monitor activity on the firewall, double-click <NEW> in Figure 409. You get now the window shown in Figure 410.



Figure 410. Add Alert Log Facility

This window provides two sections. The first one lets you set up the basic options for logging. The second one is related to archiving log files. These options are explained in 16.1.4, “Archive Log Files” on page 413. You have the following options to specify the log facility properties:

Facility The log facility determines the type and source of information that is logged (see 16.1.2, “Log Facilities” on page 410). You can't select the Admin Audit Log facility. This facility can only be created via the command line (see 16.1.5, “Manage the Admin Audit Log Facility” on page 417).

Priority Specifies the log priority level for the selected facility. The log priority levels are listed in order of increasing severity (see 16.1.1, “Logging Priority Levels” on page 409). The priority you select will be the minimum level that gets logged.

Log Filename Fill in the log filename. The log filename must have an absolute path (beginning with the drive). The path to the file must exist.

See Figure 410 on page 411 for the settings we chose for the mail facility. We decided to log at the debug priority level for the following reasons:

- We have plenty of disk space.
- We will archive regularly to efficiently use the space available.
- We need to capture everything while we test the firewall.

How much should you log?

From the time you set up the firewall to the time you finish the setup we advise you to log at the debug level or you risk missing vital information. After successful completing the testing you could decrease the log priority level.

We repeated the above steps for the Alert Log log facility.

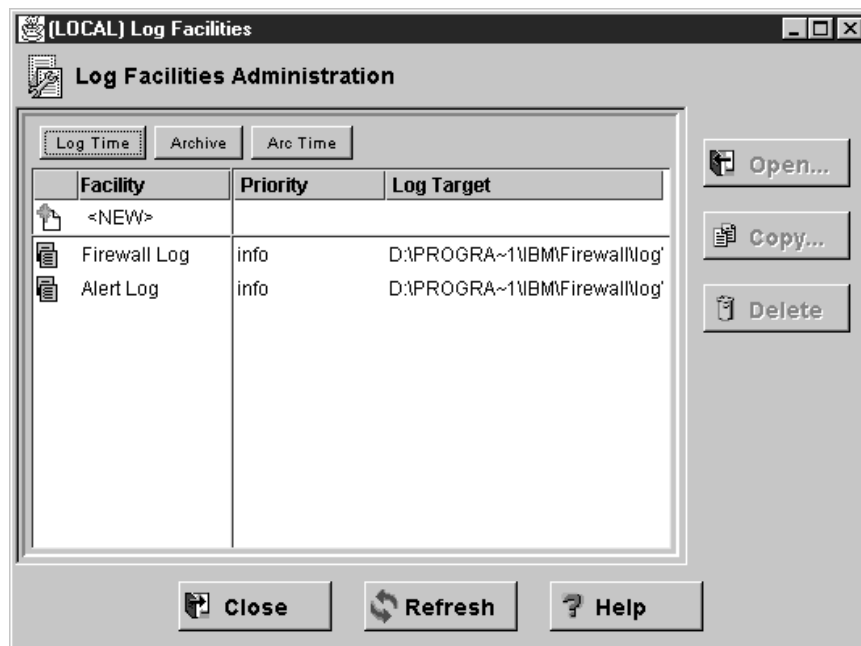


Figure 411. Completed Log Facilities

As you can see, we also changed the priority from the Firewall Log facility to debug level.

The configuration file for log facility properties is: C:\Program Files\IBM\Firewall\config\syslog.conf. The following example shows the syslog.conf file for the log facilities configured in Figure 411.

```
local4.debug C:\Program Files\IBM\Firewall\log\local4.log
local1.debug C:\Program Files\IBM\Firewall\log>alert.log
```

Figure 412. Log Facility Configuration File syslog.conf

The terms local1 and local4 in this file are coming from the UNIX environment. They are user-definable log facilities. Each line has the following format:

```
log_facility.log_priority log_file
```

Figure 413. Log Facility Configuration File Format

Where log_facility is the Log Facility type, log_priority is a valid log priority, and log_file is the file name of the log file.

16.1.4 Archive Log Files

The IBM Firewall can also manage the size of your log files, which can increase heavily over time. The size of your log files depends on:

- What priority level you are logging.
- Which filter rules are logged.
- The amount of traffic going through your firewall.

Therefore, it's a must to use the log archive capabilities of the IBM Firewall. The archival process removes qualifying records from an active log file, places them in a separate file, compacts the resulting file and places the new file into an archive directory.

Configuring the archive management involves three steps:

1. Enable archive management.
2. Start the Windows NT scheduler.
3. Add commands to the scheduler for periodic execution.

The IBM Firewall also provides an archiving API. This enables you to create your own archiving methods. Please refer to the *IBM eNetwork Firewall for Windows NT Reference Version 3 Release 3* for further details.

16.1.4.1 Enable Archive Management

To enable the archive management, double-click on one of your defined logging facilities in Figure 411 on page 412.



Figure 414. Modify Archive Properties

Now focus on the Log Management Properties. They give you the following options:

Archive Management When enabled, the log file will be processed according to the settings described below.

Days Until Archive Specifies the number of days until the log files are archived. The number of days until archive must be zero or greater.

Archive Filename Specifies the name of the file where the archived data will be written. An absolute path name must be specified.

Days Until Purge Specifies the number of days until the log files are purged. The number of days to keep the log files must be zero or greater. Log management does not count the current day when calculating the number of days to keep the file.

We enabled archive management and set our log to archive every seven days, clearing after 30 days. If you find disk space a problem, daily archiving can help.

The configuration file for the log management is C:\Program Files\IBM\Firewall\config\logmgmt.cfg.

This file is filled out by the Configuration Client. You can check here to make sure your archiving is set right. Figure 415 shows the logmgmt.cfg file for the log facilities configured in Figure 411 on page 412.

```
"C:\Program Files\IBM\Firewall\log\local4.log" 7 "C:\Program Files\ibm\firewall\log\local4.zip" 30
"C:\Program Files\IBM\Firewall\log\alert.log" 7 "C:\Program Files\IBM\Firewall\log\alert.zip" 30
```

Figure 415. Log Facility Archive Management Configuration File logmgmt.cfg

Note the double quotes, which are necessary because of the space in the first directory name. Each line has the following format:


```
log_file days_in_log_file archive_file days_in_archive
```

Figure 416. Log Management Configuration File Format

Where `log_file_name` is the name of the log file, `archive_file` is the name of the compressed archive file, `days_in_log_file` is the number of days until log entries are archived, and `days_in_archive` is number of days until entries are purged from the archive file.

You can also review your settings with the `fwlog` command. Figure 417 shows an example output.

```
C:\>fwlog cmd=list
1      facility = Firewall Log
      priority  = debug
      logfile   = C:\Program Files\IBM\Firewall\log\local4.log
      logtime   = 7
      arcfile   = c:\Program Files\ibm\firewall\log\local4.zip
      arctime   = 30

2      facility = Alert Log
      priority  = debug
      logfile   = C:\Program Files\IBM\Firewall\log\alert.log
      logtime   = 7
      arcfile   = C:\Program Files\IBM\Firewall\log\alert.zip
      arctime   = 30

3      facility = Mail
      priority  = debug
      logfile   = C:\Program Files\IBM\Firewall\log\mail.log
      logtime   = 7
      arcfile   = C:\Program Files\IBM\Firewall\log\mail.zip
      arctime   = 30

C:\>
```

Figure 417. Log File Properties Using the `fwlog cmd=list` Command

16.1.4.2 Start the Windows NT Scheduler

While updating the three log facilities you will probably see a panel similar to Figure 418.

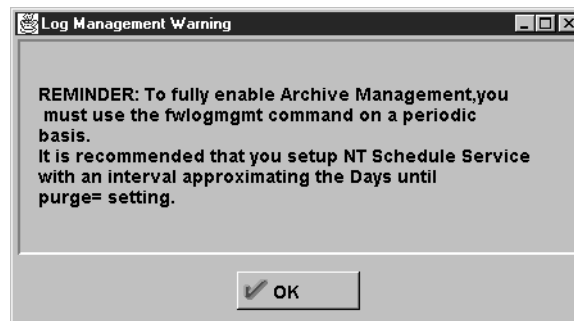


Figure 418. Archiving Warning Message

Archive management also requires that the `fwlogmgmt` command be submitted on a periodic basis.

To enable the Windows NT scheduler server open the Services Control Panel, select the **Schedule** service from the list and click on the **Startup** button to open the Service Type window. Choose the **Automatic** radio button to enable the Schedule service. Close the window by clicking **OK**. Finally start the Schedule service by clicking on the **Start** button. Your Services window should now look like Figure 419.

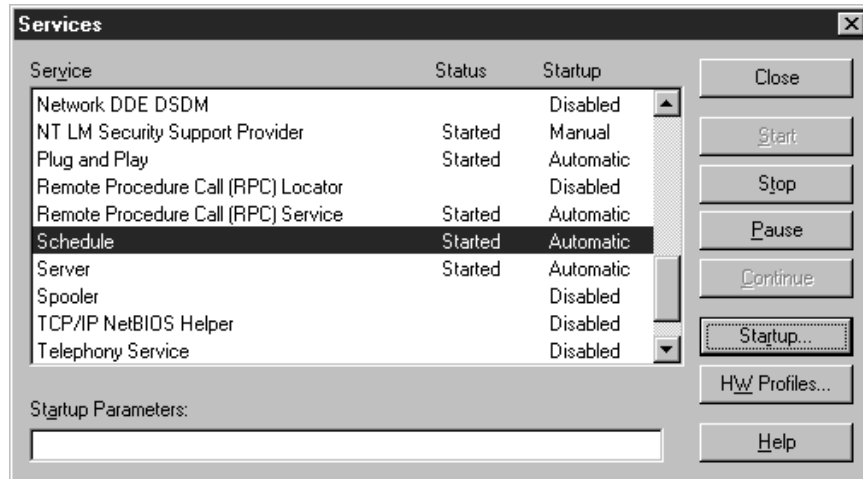


Figure 419. Enabled Windows NT Schedule Service

Unfortunately it won't stop the message in Figure 418 on page 415 from appearing, but you will be able to safely ignore it.

16.1.4.3 Add Periodic Commands

To complete configuration of archiving you need to add two commands to the Windows NT scheduler for periodic execution.

```

C:\>fwlogmgmt
fwlogmgmt -l
          -a
-a : purges the archives
-l : archives the accumulated log messages.

C:\>at 3:00 /every:M,T,W,Th,F,S,Su fwlogmgmt -l
Added a new job with job ID = 0

C:\>at 4:00 /every:M,T,W,Th,F,S,Su fwlogmgmt -a
Added a new job with job ID = 1

C:\>at
Status ID  Day          Time          Command Line
-----
          0  Each M T W Th F S Su  3:00 AM      fwlogmgmt -l
          1  Each M T W Th F S Su  4:00 AM      fwlogmgmt -a

C:\>

```

Figure 420. Adding Periodic Commands

The first command will archive your logs on every day of the week at three o'clock in the morning and the second will purge entries over the age you specified earlier, every day at four o'clock in the morning. Log file archiving is a processor- and disk-consuming task, so be sure you choose a time-frame where your firewall will probably not be very busy. Also choose different start times for both commands.

You can also use both log management commands from the command line, but we recommend you configure them as scheduled services. This decreases the chances of lost logging entries due to a full hard disk.

16.1.5 Manage the Admin Audit Log Facility

Admin Audit Log files must be managed by command line from the local machine. This is because all log files that are configured with the Configuration Client can also be browsed with it. The Admin Audit Log file contains sensitive administrative data and should therefore only be browsed by the primary firewall administrator.

Figure 421 shows an example how to create an Admin Audit Log file and change its archive properties.

```
C:\>fwlog cmd=add facility=audit priority=debug logfile="c:\program files\ibm\firewall\log\audit.log"
Command completed successfully.

C:\>fwlog cmd=change index=4 logtime=7 arctime=30 arcfile="c:\program files\ibm\firewall\log\audit.zip"
4      facility   =   Admin Audit Log
      priority    =   debug
      logfile     =   c:\program files\ibm\firewall\log\audit.log

Are you sure the above record is the correct one to process?
Please type: 1=Confirm or 2=Cancel.
1
Command completed successfully.

C:\>
```

Figure 421. Manage an Admin Audit Log File

The first command is used to create an Admin Audit Log file. If you create a log file with the command line you have to specify at least these three parameters (facility, priority, and logfile). The second command is used to enable the archive management for this log file.

16.1.6 Examine the Firewall Log Files

You can use the IBM Firewall Log Viewer to display your firewall log files. To open the Log Viewer click on the **Log Viewer** button at the bottom right of the Configuration Client. Now select one of your defined log files with the File Name pull-down menu.

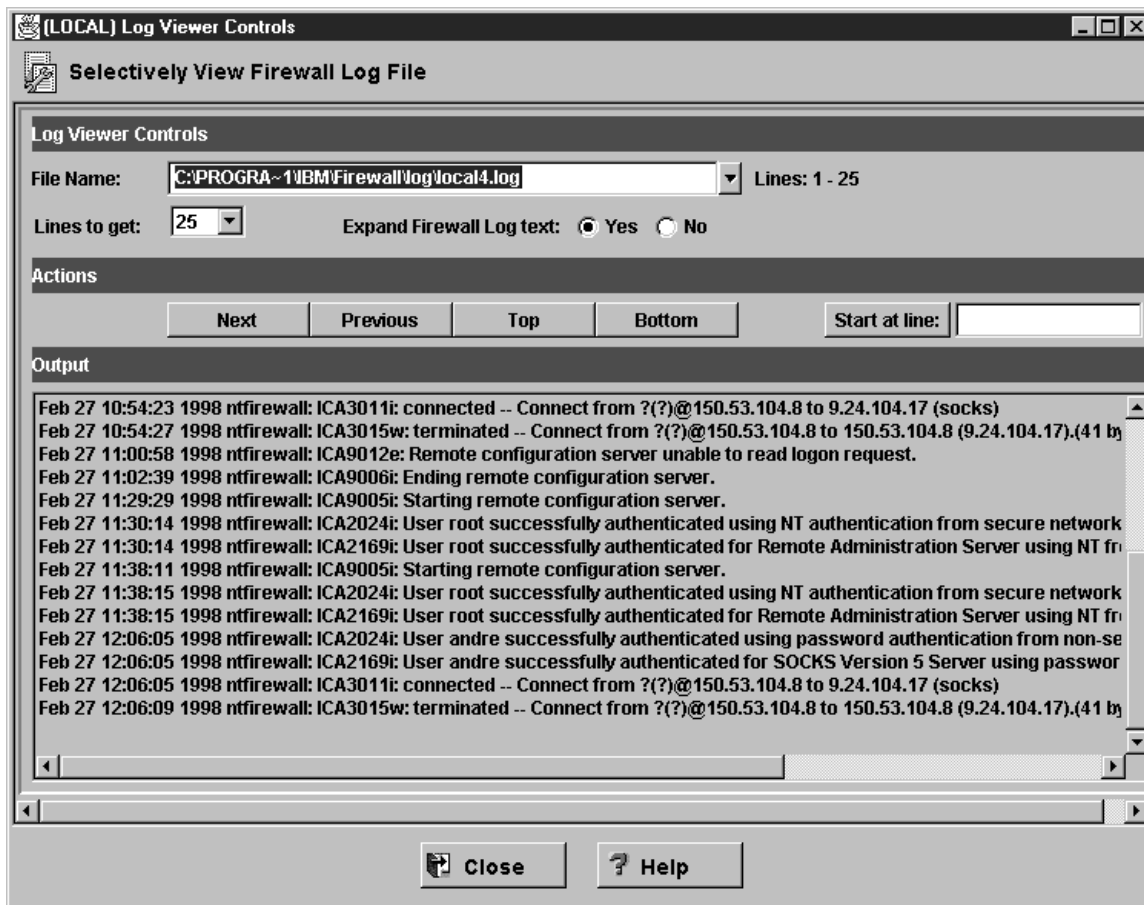


Figure 422. Log Viewer

You can navigate in the log file by clicking the **Next**, **Previous**, **Top**, and **Bottom** buttons. You can change the number of lines displayed at one time by changing the value in the Lines to get box. The Start at line box allows you to jump to a specific line given in the entry field behind the button. The Expand Firewall Log Text buttons allow you to change between a log output with and without textual messages.

Examining Large Log Files

Depending on the speed and memory of your machine examining large log files with the Log Viewer can be a boring task. So don't give up waiting if you don't receive a response immediately.

16.1.7 Configure Logging Sources

Some of the IBM Firewall services don't log by default. Therefore you have to manually enable the logging for these services if you want to receive more information.

16.1.7.1 Connection Rules

Before any denied or permitted IP packets are logged you have to enable the connection rules logging with the Connection Activation panel. To open the panel double-click on **Traffic Control** → **Connection Activation** in the navigation tree.

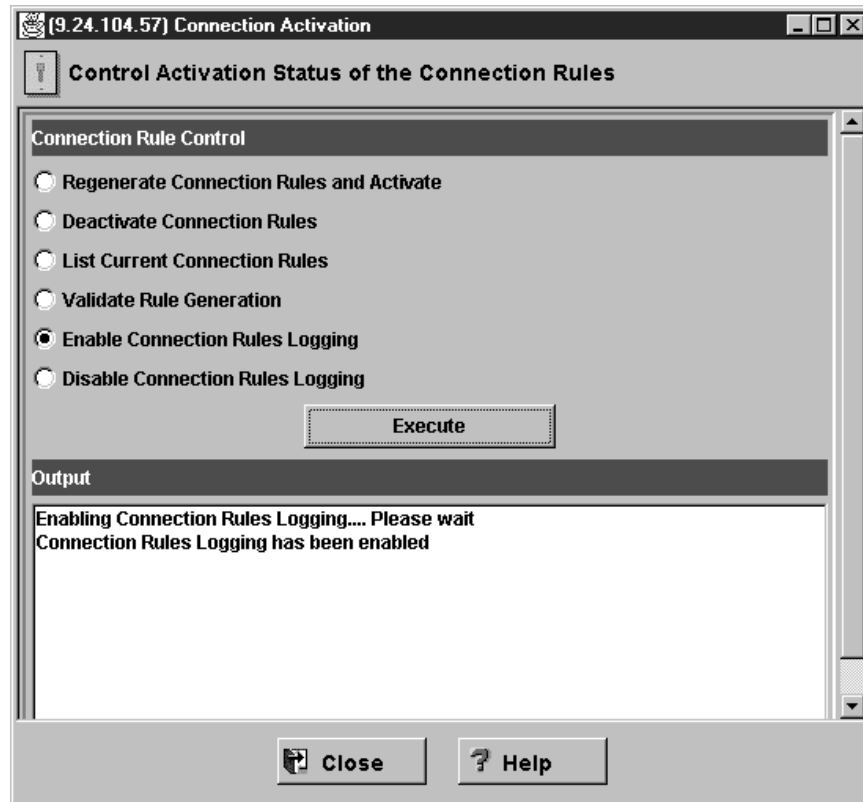


Figure 423. Enable Connection Rules Logging

Check the **Enable Connection Rules Logging** box and click on the **Execute** button. If the logging was successfully enabled, you get a message in the Output part of the panel.

Connections Rules Logging after Restart

At the time of writing, the connection rules logging is disabled after restarting the firewall. Unless you manually enable the logging, you don't see any IP packets in the Firewall Log file.

16.1.7.2 HTTP Proxy

The IBM Firewall HTTP proxy can log the processed requests. This information is useful if you are interested in the usage or misuse of this service.

To enable the HTTP log double-click the **HTTP** document in the navigation tree.

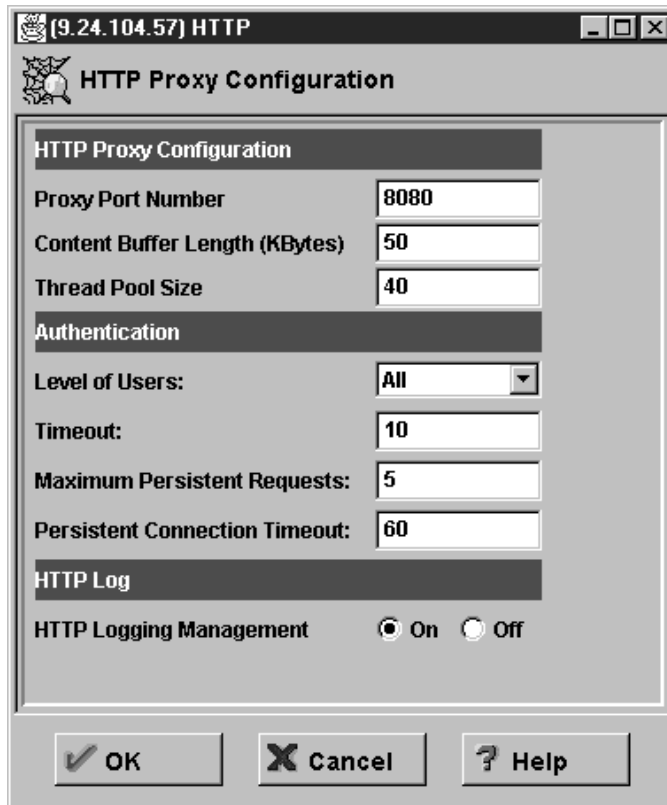


Figure 424. Enable HTTP Request Logging

Check the **On** box for the **HTTP Logging Management**. If the log management check box was Off before, the Configuration Client will ask you to restart the HTTP proxy to activate the change.

16.2 Monitoring and Alerts

There are two ways to extract important information out of the log files:

- Generating real-time alerts.
- Building analysis reports.

This section covers the generation of real-time alerts. This is the best way to begin extracting information from your logs. Once you become proficient with this facility you will be able to customize a variety of alerts to notify you in case of firewall misuse. Building analysis reports is covered by 16.4, “Building Reports” on page 436.

16.2.1 Log Monitor Thresholds

The generation of real-time alerts is based on the violation of configurable thresholds. A threshold consists of a *count* and a *time* parameter. A threshold is violated if a number of specific events (count) is exceeded in the specified amount of time (time). You can configure the following types of thresholds:

Type	Description
Total Authentication Failures Threshold	Counts all authentication failures, regardless of the originating user ID or host.
Per User Threshold	Authentication failures against any particular user ID.
Per Host Threshold	Authentication failures originating from any particular host.
Message Threshold	Occurrences of a ICA message tag in the Firewall Log log file.

You will have to experiment with the threshold settings in your environment so you don't get swamped with too many alerts, but don't lose any important information.

While this list may seem limited, the ability to set a message threshold gives you a lot of options. An overview of messages is in 16.2.3, "Alert Message Delivery Methods" on page 422. For a complete breakdown see *IBM eNetwork Firewall for Windows NT Reference Version 3 Release 3*.

16.2.2 Alert Messages

Alert messages will be delivered by the Log Monitor to keep you informed of firewall use and misuse.

The format of all ICA messages is as follows:

ICAxxxxa

Figure 425. ICA Message Tag Format

Where ICA is a fixed three-byte identifier, xxxx is the message number (see Table 37), and a is the message severity indicator (see Table 38 on page 422).

The numbers are classified into the following categories:

Numbers	Category
0000 - 0999	Intrusion alarm
1000 - 1999	Filters
2000 - 2999	Proxy-related messages.
3000 - 3999	SOCKS-related messages.
4000 - 4999	Pager-related messages
5000 - 5999	Secure Socket Layer
6000 - 6999	Virtual Private Network
7000 - 8999	Available for Future Use
9000 - 9999	General/others

There are four levels of severity, which correspond to the log priority levels as shown in Table 38 on page 422.

Message Severity Indicator	Description	Corresponding Log Priority Level
		debug
i	info	information
w	warning	warning
e	error	error
s	severe	critical

16.2.3 Alert Message Delivery Methods

Every time a threshold is violated the IBM Firewall generates a real time alert message (ICA message). The delivery of these messages can take place in four ways. You can combine the delivery methods as you want:

Type	Description
Mail Notification	Mails the ICA message to a user or a list of users.
Execute Command	Executes a user-defined command with the ICA message as the first parameter.
Pager Notification	Pages a message to a defined pager.
Log Entry	Logs the ICA message in an Alert Log file. This file is displayed in the Alerts Display from the Configuration Client.

16.2.4 Log Monitor Administration

Open the **System Administration** → **System Logs** → **Log Monitor Thresholds** document from the Configuration Client navigation tree. In Figure 426 on page 423 you see a list of four predefined log monitors.

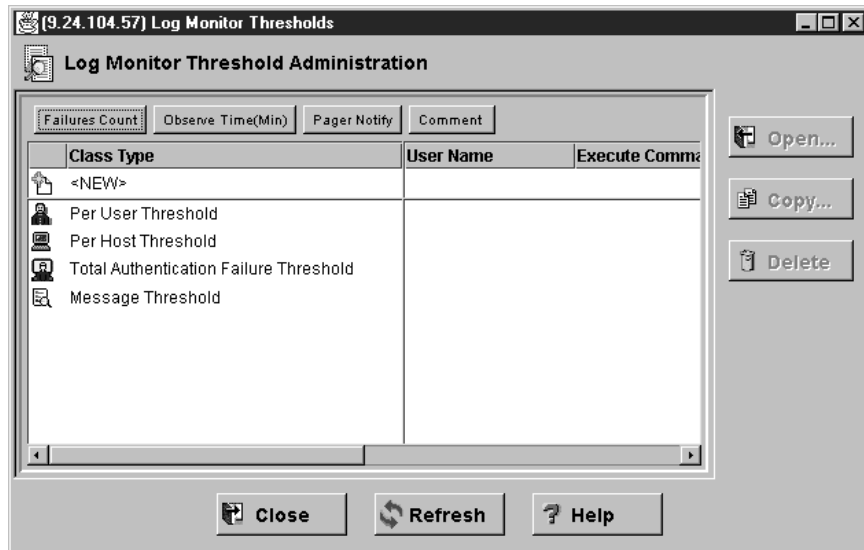


Figure 426. Log Monitor Threshold Administration

You can open the Add Log Monitor panel by double-clicking the **<NEW>** entry in the list shown in Figure 426.

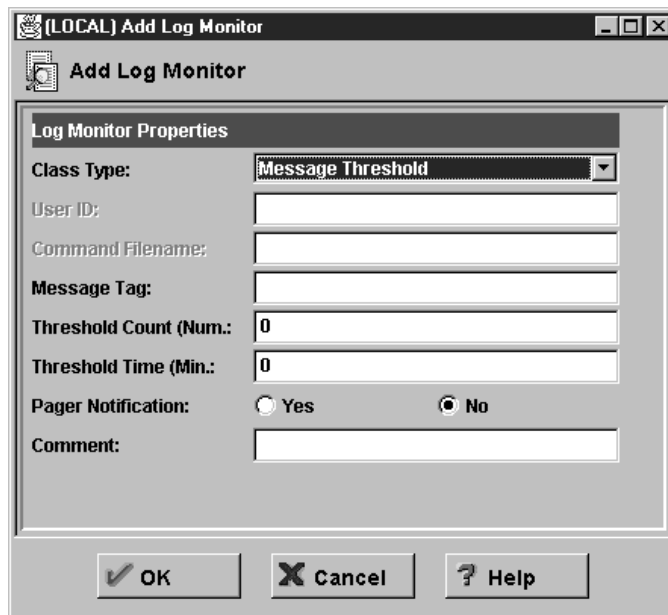


Figure 427. Add Log Monitor

This dialog allows you to define new threshold monitors (see 16.2.5, “Configure Threshold Monitors” on page 424) and new alert message delivery monitors (see 16.2.6, “Configure Delivery Monitors” on page 426). With the Class Type field you indicate the type of monitor. Click the pull-down menu to choose from the list of available options. This list contains entries to define threshold monitors (see Table 36 on page 421) and entries to specify delivery monitors (see Table 39 on page 422). You can’t select the delivery monitor types Log Entry and Pager Notification with this menu.

The other fields are explained in the corresponding sections. Be aware that some fields are not available for all monitors.

16.2.5 Configure Threshold Monitors

Threshold monitors contain a type and a threshold. In case of threshold violation they trigger an alarm message. As we discussed in 16.2.1, “Log Monitor Thresholds” on page 420 there are four types of thresholds:

- Total Authentication Failures Threshold
- Per User Threshold
- Per Host Threshold
- Message Threshold

The first three are authentication failure threshold monitors and they have the same options. Therefore we describe only the Total Authentication Failure Threshold monitor and the Message Threshold monitor. You should check the three predefined threshold monitors in Figure 426 on page 423 and change them if necessary.

There is also one predefined message tag threshold monitor, but you can define as many message tag monitors as you want. You find a complete list of message tags in the *IBM eNetwork Firewall for Windows NT Reference Version 3 Release 3*. A list of recommended message tag is provided by Appendix C, “Recommended ICA Message Tags” on page 607.

16.2.5.1 Configure Authentication Failure Monitors

Double-click the **Total Authentication Failure Threshold** monitor in Figure 426 on page 423. The Modify dialog opens.

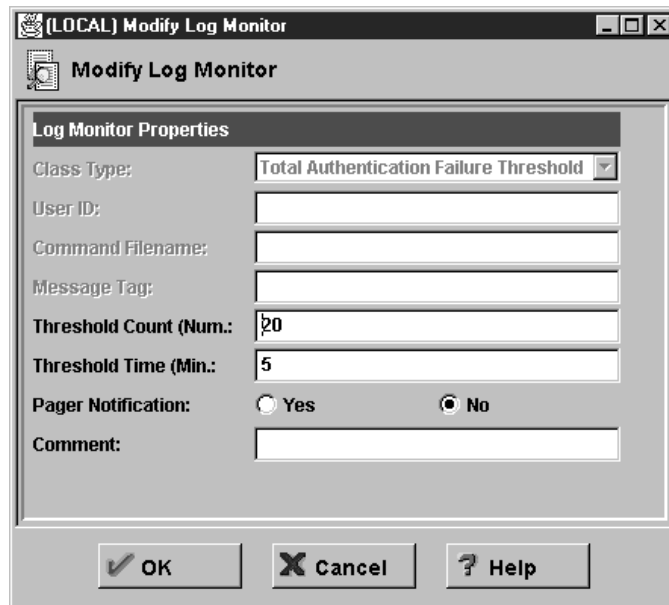


Figure 428. Total Authentication Failure Threshold Monitor

You have the following options:

Threshold Count (Num.) Specifies the maximum number of occurrences of a specified log message tag. If the number of occurrences exceeds the threshold count within the specified time (in minutes), the log monitor sets off an alarm. Threshold count can't be greater than 99999.

Threshold Time (Min.) Specifies the time period that defines when a certain number of events has exceeded the threshold. If the threshold has been exceeded, the log monitor sets off an alarm. Threshold time can't be greater than 99999. A value of 0 for time indicates an unlimited time period.

Pager Notification Enables the pager notification for this monitor when the threshold is exceeded.

Comment Contains an optional textual description of the monitor.

To create new authentication threshold monitors double-click **<NEW>** in Figure 426 on page 423. Choose one of the three authentication threshold monitors from the Class Type pull-down menu and fill out the other fields.

16.2.5.2 Configure Message Tag Monitors

You can define a new message tag threshold monitor by double-clicking **<NEW>** in Figure 426 on page 423.

The screenshot shows a Windows-style dialog box titled "(9.24.104.57) Add Log Monitor". Inside, there's a tab labeled "Add Log Monitor". Below that is a section titled "Log Monitor Properties". The "Class Type" is set to "Message Threshold" in a pull-down menu. Other fields include "User ID", "Command Filename", "Message Tag" (containing "ICA1032"), "Threshold Count (Num.)" (1), "Threshold Time (Min.)" (1), "Pager Notification" (with "No" selected), and "Comment" (containing "Filter Rules Updated"). At the bottom are "OK", "Cancel", and "Help" buttons.

Figure 429. Message Tag Threshold Monitor

Choose **Message Threshold** from the Class Type pull-down menu. Now you have the following options:

Message Tag You specify here the identifying message number. A list of all messages, including associated tags and descriptions, is provided in the *IBM eNetwork Firewall for Windows NT Reference Version 3 Release 3*. A list of recommended messages is provided by Appendix C, "Recommended ICA Message Tags" on page 607.

Threshold Count (Num.) Specifies the maximum number of occurrences of a specified log message tag. If the number of occurrences exceeds the threshold count within the specified time (in minutes), the log monitor sets off the alarm. Threshold count can't be greater than 99999.

Threshold Time (Min.) Specifies the time period that defines when a certain number of events has exceeded the threshold. If the threshold has been exceeded, the log monitor sets off an alarm. Threshold time can't be greater than 99999. A value of 0 for time indicates an unlimited time period.

Pager Notification Enables the pager notification for this monitor when the threshold is exceeded.

Comment Contains an optional textual description of the monitor.

16.2.6 Configure Delivery Monitors

As we discussed in 16.2.3, "Alert Message Delivery Methods" on page 422 there are four alert message delivery monitors.

- Mail Notification
- Pager Notification
- Execute Command
- Log Entry

You can use them to keep you informed of threshold violations from the firewall.

16.2.6.1 Configure Mail Notification Monitors

You can define a mail notification monitor by double-clicking **<NEW>** in Figure 426 on page 423.

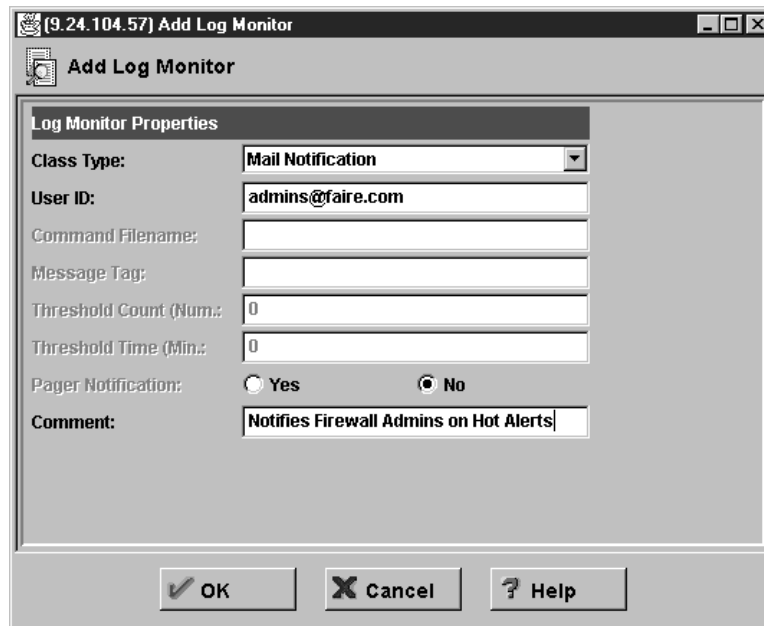


Figure 430. Mail Notification Delivery Monitor

Choose **Mail Notification** from the Class Type pull-down menu. Now you have the following options:

User ID Specify here the e-mail address of the recipient.

Comment Contains an optional textual description of the monitor.

Mailing Lists

You can define multiple mail notifications but we recommend you define only one. Create a mailing list for your firewall administrators at your internal mail server and use the address of this list as the recipient. Now you can easily configure this list in case of holidays or the exchange of an administrator.

16.2.6.2 Configure Pager Notification

The pager notification is specified on a per threshold monitor basis. You can decide for every threshold monitor whether you want to send a page or not in case of a violation of this monitor. But you can only define one fixed message that is sent to you in case of an alert. See 16.2.6.3, “Configure Command Execution Monitors” for an example to overcome that limitation.

See 16.2.5, “Configure Threshold Monitors” on page 424 for details on how to enable paging for a threshold monitor. You will find some details about pager setup in 16.2.7, “Pager Setup” on page 431.

16.2.6.3 Configure Command Execution Monitors

You can define a command execution monitor by double-clicking <NEW> in Figure 426 on page 423.

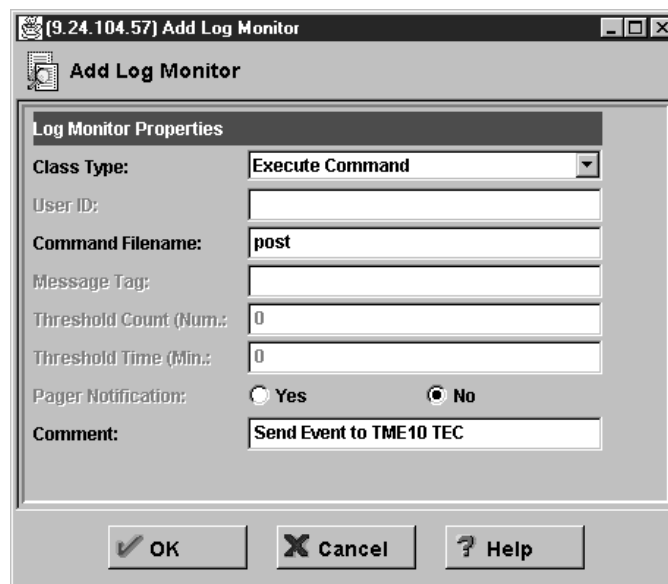


Figure 431. Command Execution Monitor

Choose **Execute Command** from the Class Type pull-down menu. Now you have the following options:

Command Filename Indicates the file name of the program that will be executed when a threshold is exceeded.

Comment Contains an optional textual description of the monitor.

Serializing

The program you specify is launched in a new process every time an alert is generated and it's possible that multiple instances of the program can be running at the same time. So the program is responsible for serializing access to shared resources.

A descriptive alert message is sent to the program. This enables you do advanced functions, for example:

- Execute programs depending on the ICA message tag.
- Forward messages to your systems management environment.
- Automatic update of your rule base to disable logging of specific packets.

We show you now two examples of using the command execution monitor:

- Forward messages to a Tivoli Enterprise Console (TEC) server.
- Forward messages to a pager with a detailed description.

The first example shows you how to forward messages to a TEC server using the `postmsg.exe` command. This command is part of the TEC Adapter for NT.

You first have to transfer the `postmsg.exe` file to your firewall and place it into a directory which is on your PATH variable. We use the `C:\WinNT\system32\` directory. Within the same directory create a command file (`post.cmd`). We use this command file to execute `postmsg.exe` with the appropriate parameters in case of threshold violations (see Figure 431 on page 427).

```
postmsg -S 9.24.104.249 -m "%*" FWNT_Alert NT
```

Figure 432. Command Execution Monitor Example (`post.cmd`)

The parameters are the TEC server (`-S 9.24.104.249`), the message (`-m "%*"`), a user-defined class (`FWNT_Alert`) and the source (`NT`).

Within the TEC server you have to create the BAROC file which contains the `FW_Alert` class.

```
TEC_CLASS :
    FWNT_Alert ISA NT_Base
    DEFINES {
        severity: default="CRITICAL";
        sub_source: default="FWNT";
    };
END
```

Figure 433. BAROC Class `FWNT_Alert`

Please see 21.6, "TME 10" on page 576 or your *TME 10 Enterprise Console User's Guide* for how to update your rule base accordingly.

Finally you have to create a connection between the firewall secure interface and your TEC server using a service with IP filter rules for the ports 111/udp and 1229/tcp. These ports are used by the `postmsg.exe` command to communicate with the TEC server.

Figure 434 on page 429 shows some sample events send to the TEC server.

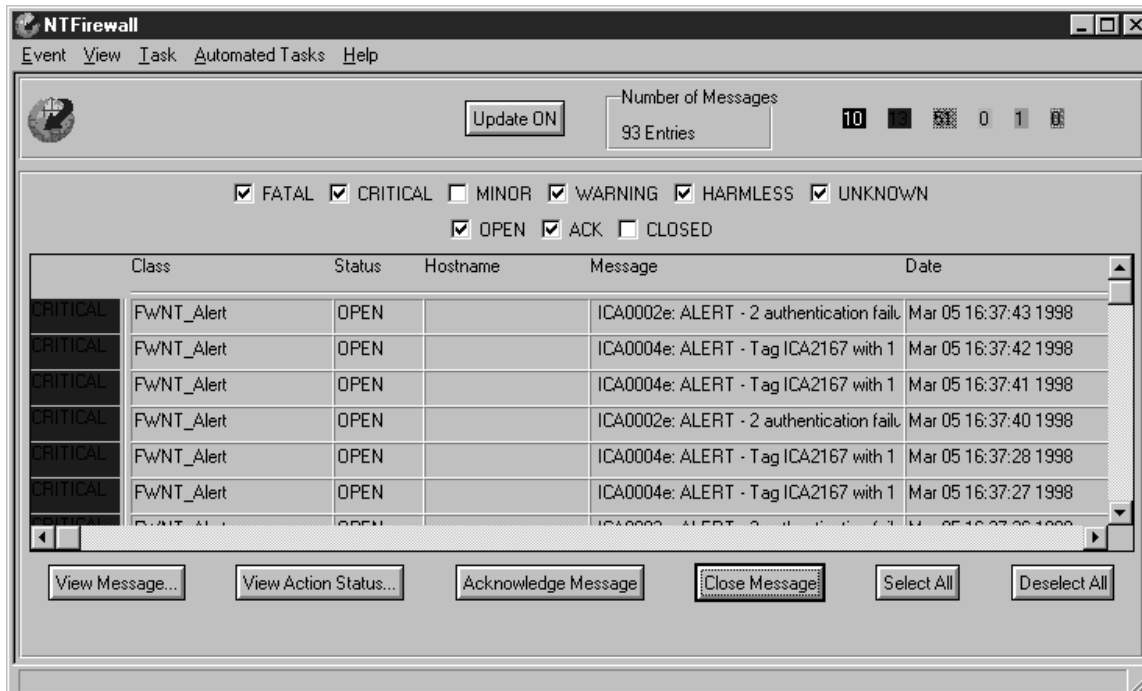


Figure 434. Tivoli Enterprise Console with Firewall Alerts

You find another, more detailed, approach to monitor the IBM Firewall with TME 10 in 21.6, “TME 10” on page 576.

With the second example we will show you how to send a message to a pager containing the specific ICA message that triggers the monitor.

To implement this we use pager.exe, which is part of the IBM Firewall distribution. You can also use this command to test your pager setup (see 16.2.7.4, “Test Pager” on page 433). The configuration file used for this program is pager.cfg. Refer to 16.2.7, “Pager Setup” on page 431 to set up your pager with the Configuration Client. Figure 435 shows the sendpage.cmd file, which is executed from a command execution monitor.

```
pager msg="%*"
```

Figure 435. Sending Pages with sendpage.cmd

The message is passed as parameters to sendpage.exe. We use %* to get the whole message. You can also extend the batch program to do different things depending on the ICA message tag.

Place the sendpage.cmd file into a directory that is on your PATH variable. We use the C:\WinNT\system32\ directory. Finally create the following command execution monitor:

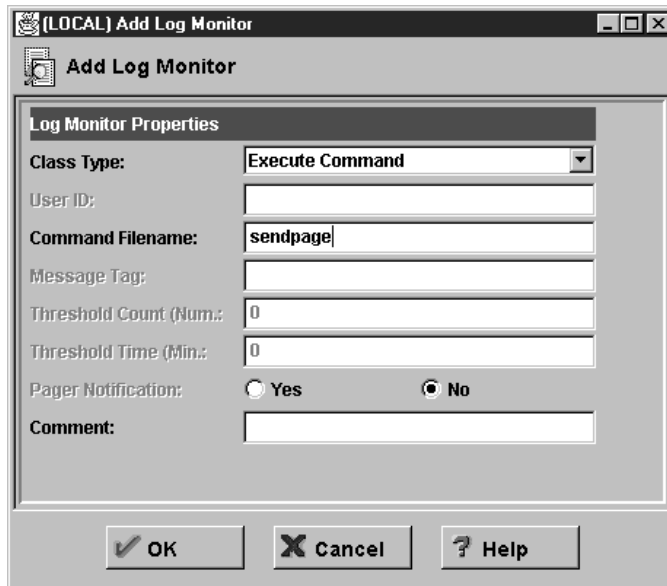


Figure 436. Command Execution Monitor

16.2.6.4 Configure Log Entry Notification

The logging of alerts in a special log file takes place if you create an Alert Log facility (see 16.1.3, “Manage Log Facilities” on page 410). Once you have created this facility all violations of defined thresholds are displayed in the Alerts Display of the IBM Firewall Configuration Client.

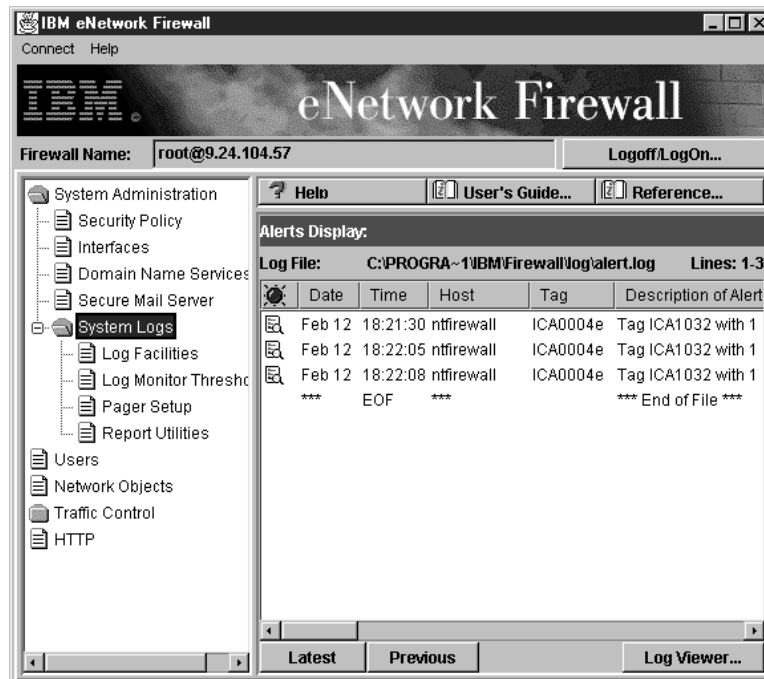


Figure 437. Alerts Display in the Configuration Client

You can get the latest alerts by clicking on the **Latest** button in the bottom area. To see old entries use the **Previous** button.

16.2.7 Pager Setup

In this section we describe how to configure a pager to work with the IBM Firewall. You must configure three components:

- Command customization
- Carrier administration
- Modem administration

Before starting, you need to make sure that you have a carrier where you can dial-in and send pages. Your carrier should be able to page to the locations where your administrators are. You also need to get the correct modem phone numbers, pager ID, and modem parameters from your carrier. The carrier must support the TAP protocol. TAP stands for Telocator Alphanumeric Protocol. It is an industry-standard protocol for sending a page via a modem. See

http://www.mot.com/MIMS/MSPG/pcia_protocols/tap_v1p8/index.html

for details on this protocol.

Windows NT Modem Setup

You must not set up your modem for Windows NT. You can do all the necessary configuration with the IBM Firewall Configuration Client. This also makes sure that your modem is used for outgoing calls only.

The Windows NT Telephony Service is disabled during the firewall installation. Therefore, you can't set up any dialing properties within NT.

16.2.7.1 Command Customization

To configure the command that is sent to the pager if a threshold is violated, open the **System Administration** → **System Logs** → **Pager Setup** document in the navigation tree.

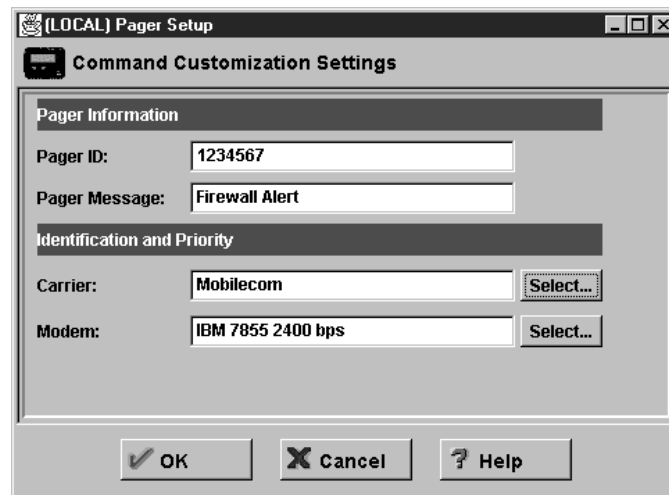


Figure 438. Pager Customization

You can specify the following options:

Pager ID This is usually a unique PIN number assigned to your pager by your carrier company.

Pager Message Specifies the pager message that appears on the pager when the log facilities issue the page. Do not use double quotes at the start and end of your message.

Carrier Name Click on **Select** to select or define a carrier.

Modem Name Click on **Select** to select or define a modem.

You can only specify one message for your defined threshold monitors. For numeric pagers, this must be a number only. For alphanumeric pagers, this can be a text message. Do not exceed the maximum message length for alphanumeric pagers or your message might be truncated.

16.2.7.2 Carrier Administration

If you click the **Select** carrier button in Figure 438 on page 431, you get the Carrier Administration window.

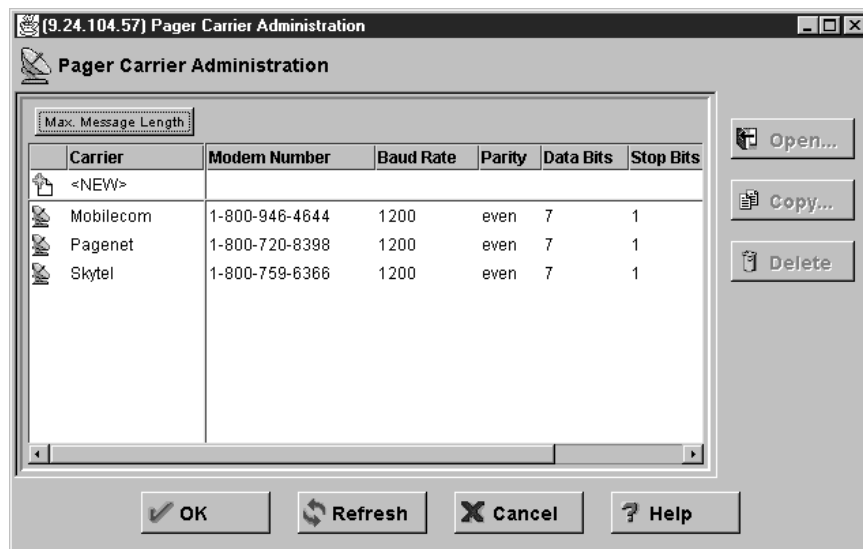


Figure 439. Carrier Administration

Select one of the predefined carriers or define a new one by double-clicking <NEW>.

16.2.7.3 Modem Administration

If you click the **Select** modem button in Figure 438 on page 431, you get the Modem Administration window.

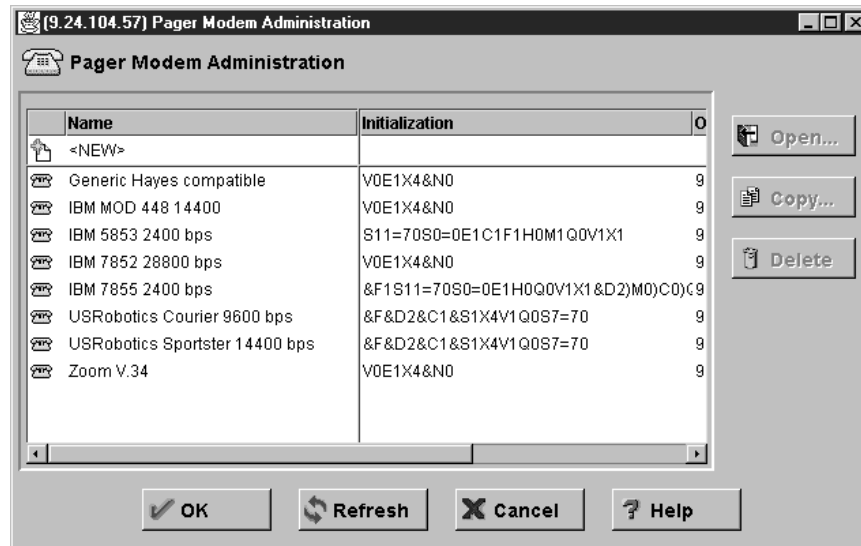


Figure 440. Modem Administration

Select one of the predefined modems or define a new one by double-clicking <NEW>.

It is quite possible you will need to adjust the init string to achieve proper communication between your modem and your Carrier's modem.

16.2.7.4 Test Pager

Be sure you have successfully tested the transmission of pages in case of threshold violations before going into production. You can use the pager command, which is part of the IBM Firewall distribution, to send a message to a pager.

The pager command has four parameters. The format is shown in Figure 441.

```
pager id="pager_id" carrier="carrier_name" modem="modem_name" msg="message"
```

Figure 441. Format of the Pager Command

All parameters are optional. Defaults are obtained from the pager configuration file (pager.cfg), which in turn references entries in the modems.cfg and carriers.cfg files. All files are located in the C:\Program Files\IBM\Firewall\config\ directory. The quotes are only needed if you use characters such as blanks, which have a special meaning to the command shell.

16.3 Log File Formats

You need to understand the log files and their format if you want to extract further information from them. As described in 16.1.2, "Log Facilities" on page 410 there are three Log Facilities using their own formats:

- Firewall Log
- Alert Log
- Admin Audit Log

The format of the Windows NT Firewall Log file is not compatible with the AIX format. This means that if you want to use report utilities which are based on the

AIX Firewall Log format, you will need to convert your files to the AIX format first. You can use the `fwlogcvrt` command to do so.

16.3.1 Firewall Log Format

The entries in the Firewall Log files have the following format:

```
date time fw_name:year;pid: Amsg_num;msg_ID;var_1;...;var_n;
```

Figure 442. Firewall Log File Format

The fields are explained in Table 40.

Field	Description
date	Month and day when entry occurs. The format is MMM DD.
time	Time when entry occurs. The format is HH:MM:SS.
fw_name	The name of the firewall that logged this entry.
year	year is the four-character year.
pid	pid is the process ID to which the entry applies.
Amsg_num	msg_num is a sequential integer that the Report Utilities use to access the appropriate, translated message text from the <code>fw_log.cat</code> file. The number is preceded by a log level indicator letter. This indicator distinguishes both the platform that originated the log entry and any differences in log format.
msg_ID	msg_ID is the external number of the message, such as ICA0001e (see 16.2.2, "Alert Messages" on page 421).
var_n	var_1 to var_n represents the values of message variables, where n is the number of variables in the message definition.

Figure 443 shows a few example entries in the short form.

```
Mar 01 09:36:18 ntfirewall:1998;119: A2067;ICA1033i;3;2;09:36:18;Mar-01-1998;
Mar 01 09:37:51 ntfirewall:1998;275: A2071;ICA1035i;enabled;09:37:51;Mar-01-1998;
Mar 01 09:37:54 ntfirewall:1998;85: A2073;ICA1036i;#:55;R:d; i::9.24.104.57;s::9.24.104.70;d::150.53.104.8;
p::tcp;sp::2059;dp::110;r:r;a:s;f:n;T:0;e:n;l:44;
```

Figure 443. Firewall Log Example Entries

You can also convert this log into a more readable form using the `fwlogtxt` command:

```
C:\>fwlogtxt < C:\Program Files\IBM\Firewall\log\local4.log
Mar 01 09:36:18 1998 ntfirewall: ICA1033i: Filter support (level 3.2) initialized at 09:36:18 on Mar-01-1998
Mar 01 09:37:51 1998 ntfirewall: ICA1035i: Status of packet logging set to enabled at 09:37:51 on Mar-01-1998
Mar 01 09:37:54 1998 ntfirewall: ICA1036i: #:55 R:d i:9.24.104.57 s:9.24.104.70 d:150.53.104.8 p:tcp
sp:2059 dp:110 r:r a:s f:n T:0 e:n l:44

C:\>
```

Figure 444. Converted Firewall Log Example Entries

16.3.2 Alert Log Format

The entries in the Alert Log files have the following format:

```
date time fw_name:year;pid: msg_num;msg_ID: descr
```

Figure 445. Alert Log File Format

The fields are explained in Table 41.

Field	Description
date	Month and day when entry occurs. The format is MMM DD.
time	Time when entry occurs. The format is HH:MM:SS.
fw_name	The name of the firewall which logged this entry.
year	year is the four-character year.
pid	pid is the process ID to which the entry applies.
Amsg_num	msg_num is a sequential integer that the Report Utilities use to access the appropriate, translated message text from the fw_log.cat file. The number is preceded by a log level indicator letter. This indicator distinguishes both the platform that originated the log entry and any differences in log format.
msg_ID	Contains one of the four threshold violation message IDs (ICA0001, ICA0002, ICA0003, or ICA0004).
descr	Shows the message tag which has been violated and the number of occurrences.

Figure 446 shows a few example entries.

```
Mar 02 11:30:56 ntfirewall:1998:82; A8;ICA0004e: ALERT - Tag ICA2167 with 2 log entries.
Mar 02 11:31:11 ntfirewall:1998:82; A8;ICA0001e: ALERT - 3 authentication failures.
Mar 02 11:31:14 ntfirewall:1998:82; A8;ICA0002e: ALERT - 2 authentication failures for user sven.
```

Figure 446. Alert Log Example Entries

16.3.3 Admin Audit Log Format

The entries in the Admin Audit Log files have the following format:

```
date time fw_name:user_ID;action; var_1=a ... var_n=z[;rc=n]
```

Figure 447. Admin Audit Log File Format

The fields are explained in Table 42.

Field	Description
date	Month and day when entry occurs. The format is MMM DD.
time	Time when entry occurs. The format is HH:MM:SS.
fw_name	The name of the firewall which logged this entry.

<i>Table 42 (Page 2 of 2). Admin Audit Log Entry Fields</i>	
Field	Description
user_ID	The administrator user ID which performs the action.
action	The administrative action performed.
var_n=x	Shows the settings of the variables used for the action. The format is var_name=value.
rc=n	The return code of the action. This parameter is optional.

Figure 448 shows a few example entries.

```

Mar 02 09:31:41 ntfirewall:root;fwListProxyUser; type=long username=root logonmode=host
admin_userid=root;rc=1
Mar 02 09:31:42 ntfirewall:root;fwListLog; logonmode=host admin_userid=root;rc=1
Mar 02 09:31:43 ntfirewall:root;fwBrowseFile; cmd=browse
filename=C:\Program Files\IBM\Firewall\log>alert.log
type=literal +++4555=1 length=-17 trans=0 logonmode=host admin_userid=root;
rc=763

```

Figure 448. Admin Audit Log Example Entries

16.4 Building Reports

The IBM Firewall allows you to log the different events happening in your firewall. For example, you can log denied or permitted IP traffic, SOCKS and proxy usage, and mail events.

As we have said, there are two ways to extract important information out of the log files:

- Generating real-time alerts.
- Building analysis reports.

Generating a real-time alert is covered by 16.2, “Monitoring and Alerts” on page 420. You should also build useful reports from your firewall logging. The reports may provide information about attack rates and types of attacks.

You may also be interested in information of resource usage from the secure network. For example, information about the amount and duration of internet traffic. Or total number of sessions per given period, regardless what session type or total number of bytes transferred by FTP per given period and so on.

The firewall log files contain this information but it is not easily derived from them. With the Report Utilities it is possible to convert the firewall log files into import files for database managers such as DB2/6000, DB2, or Oracle. In this way you can use all the power of the Structured Query Language (SQL), or other tools like IBM's Visualizer or Query Management Facility to query the data and generate reports.

In 21.1, “IBM DB2” on page 517 we describe a method to import the log files into database tables from DB2 and create advanced reports. You could to build these reports with a specialized tool like TELEMATE.Net.

16.4.1 Report Utilities

You can use the IBM Firewall Report Utilities to create full text log files or import files for databases out of log files from the Firewall Log log facility. We describe here only the way to create the files necessary for creating reports with DB2. Please refer to the *IBM eNetwork Firewall for Windows NT User's Guide Version 3 Release 3* and the *IBM eNetwork Firewall for Windows NT Reference Version 3 Release 3* for usage of the Report Utilities for other purposes.

The Report Utilities can be used with the Configuration Client or via command line. The command line programs are also installed with a Configuration Client on a remote machine.

Where to Use?

We recommend that you transfer the log files to a remote machine and run the Report Utilities from the command line. Don't forget to create a connection from the firewall to your remote machine for the file transfer.

The Report Utilities consists of three programs and three SQL scripts. The program directory is automatically added to your PATH variable. You find the sample SQL scripts in C:\Program Files\IBM\Firewall\sample\report\.

Table 43. IBM Firewall Report Utilities

File	Description
fwlogcvt	Program to convert a Windows NT firewall log files to AIX firewall log files.
fwlogtxt	Program to generate full-text messages from a firewall log file.
fwlogtbl	Program to generate database import files, in DEL (delimited) format, from a firewall log file.
fwschema.ddl	File of SQL Data Definition Language (DDL) statements, suitable for defining the database tables.
fwimport.dat	File of DB2 import statements, suitable for importing the DEL files into the database tables.
fwqrysmp.dml	File of SQL Data Manipulation Language (DML) statements, suitable for generating sample reports.

The SQL scripts are specific to the DB2 family, but you can modify them for other database managers.

16.4.1.1 Using the Configuration Client

To create the DB2 import files with the Configuration Client select **System Administration** → **System Logs** → **Report Utilities** from the navigation tree. You will see the window of Figure 449 on page 438.

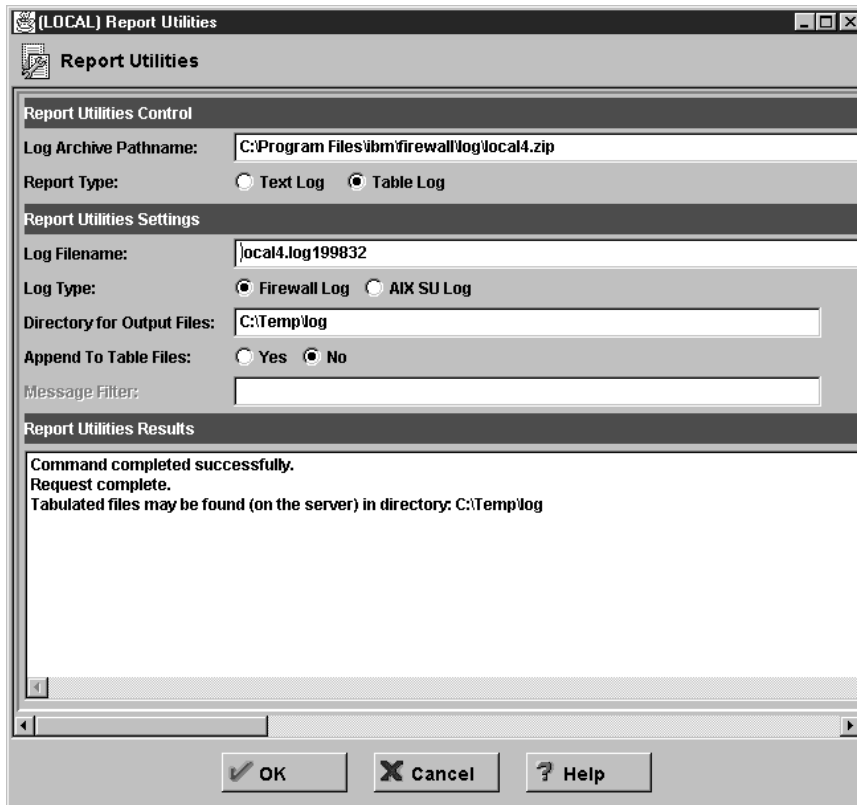


Figure 449. Report Utilities

The different fields are:

Log Archive Pathname The log archive is the directory that contains compressed log files. In the log archive field enter the directory that you specified in the archive directory field on the Log Facilities panel. Enter the absolute path name to the archive directory. If you want to view a log file that is not archived, leave this field blank.

Report Type To produce the expanded log message text, select Text Log. To create tabulated files for DB2 usage, select Table Log.

Log Filename The log filename is any one of the compressed archived log files or other valid firewall logs. If you made an entry in the log archive directory field, you can select the button in the Log Filename field to choose which log to work with. If you do not enter a log archive, the log file name you enter here must be the name of a valid, uncompressed firewall log file. You must specify a full path.

If you change the **Report Type** radio button to **Table Log** you have the following options.

Log Type Select the log type.

Directory for Output Files The Report Utilities create a bunch of files that can be imported to DB2. Specify here an output directory for these files.

Append To Table Files Select **Yes** to append the results of a table log request to existing tabulated files or **No** to replace the existing files.

Finally click on **OK** to run the Report Utilities. After successful completion you find the import files in the directory specified. Please refer to 21.1, “IBM DB2” on page 517 on how to use these files with DB2 to create reports for the IBM Firewall.

16.4.1.2 Using the Command Line

The import files can also be generated from the command line using the following command:

```
C:\>fwlogtbl -w -d c:\temp\log c:\temp\loca14.log
C:\>
```

Figure 450. Creating Import Files Using the Command Line

The parameter `-w` specifies that eventually existing output files should be replaced. `-d` is followed by the output directory. And the last parameter is the input log file. Please remember, you can use Firewall Log files only.

Please refer to 21.1, “IBM DB2” on page 517 on how to use these files with DB2 to create reports for the IBM Firewall.

16.4.1.3 Database Import Files

After using the Report Utilities either with the Configuration Client or via the command line you find a bunch of files created in your output directory. The resulting files are in delimited ASCII (DEL) file format, with no character string delimiters, and using semicolon (;) as the column delimiters. The extension for these files is `.tbl`.

The following table shows the relation between the files and the tables created by the SQL scripts.

<i>Table 44 (Page 1 of 2). Report Utilities Database Tables</i>		
Database Table	Import File	Description
ADMIN_ALERT	a_alert.tbl	Messages related to intrusion alerts.
FILTER_ACTIVE_RULE	f_rule.tbl	Active IP filter rules.
FILTER_INFO	f_info.tbl	Error or general information messages related to IP filters.
FILTER_MATCH	f_match.tbl	Matched IP filter rules.
FILTER_STATUS	f_stat.tbl	Information on status changes of IP filters.
PAGER_INFO	pgr_info.tbl	Information related to the paging feature of the Firewall for those pager messages that are mapped to the database.
PROXY_FTP	p_ftp.tbl	FTP action information from FTP sessions.
PROXY_HTTP	p_http.tbl	HTTP action information from proxy sessions.
PROXY_INFO	p_info.tbl	Error or general information messages related to proxy.

Table 44 (Page 2 of 2). Report Utilities Database Tables

Database Table	Import File	Description
PROXY_LOGIN	p_login.tbl	Information (primarily regarding authentication) about successful proxy logins.
PROXY_STATUS	p_stat.tbl	Proxy status information.
SERVER_INFO	srv_info.tbl	Information about Configuration Server status and activities.
SESSION	session.tbl	SOCKS and proxy session start/stop information.
SOCKS_FTP	s_ftp.tbl	SOCKS FTP action information from FTP sessions.
SOCKS_INFO	s_info.tbl	Error or general information messages related to SOCKS.
SSL_INFO	ssl_info.tbl	Information about SSL status and activities.
TUNNEL_CONTEXT	t_cntxt.tbl	Active tunnel context specifications.
TUNNEL_POLICY	t_policy.tbl	Tunnel policy statements.
TUNNEL_STATUS	t_stat.tbl	Information on status changes of tunnels.

Please refer to 21.1, “IBM DB2” on page 517 on how to use these files with DB2 to create reports for the IBM Firewall.

16.5 Using SOCKS Traffic Monitor

Together with the IBM Firewall you receive a SOCKS Traffic Monitor application. This application allows you to monitor the current SOCKS traffic on the firewall in real-time. To use this application you must first start the IBM Firewall SOCKS Monitor Service on your firewall. You find a description of how to start services in 13.2, “Managing Proxy and SOCKS Services” on page 306.

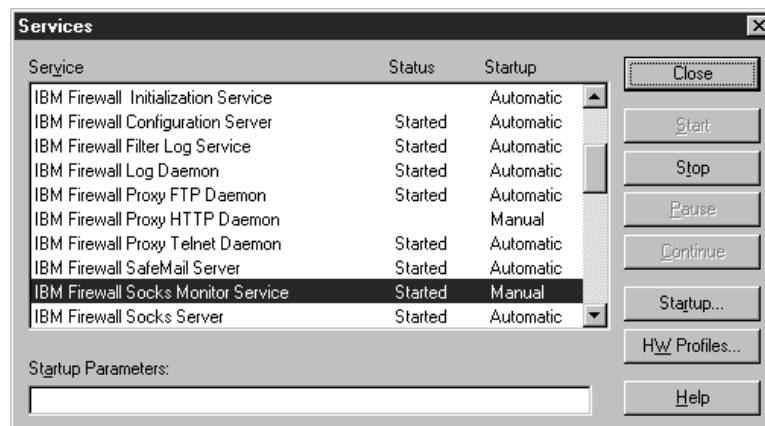


Figure 451. Started IBM Firewall SOCKS Monitor Service

Once the service is started you can start the Traffic Monitor application either on the firewall itself or on a remote client. Don't forget to define a connection for this

traffic if you want use it from a remote machine (see Chapter 8, “IBM Firewall Rule Base” on page 143 and Chapter 9, “Examples of Rules for Specific Services” on page 161).

16.5.1 Connection for Remote Usage

To use the SOCKS Traffic Monitor from a remote machine you have to define a connection from this machine to the firewall for the IP traffic.

Unfortunately there is no predefined service which we can use for that connection. Therefore, we build this connection from the scratch starting with the two rules (see Chapter 8, “IBM Firewall Rule Base” on page 143 for how to create rules and services). We only show the important windows, but don't explain the whole process.

The SOCKS Traffic Monitor uses port 5051/tcp. We start with the rule for the incoming traffic.

The screenshot shows a dialog box titled "(LOCAL) Add IP Rule" with a sub-header "Add a Rule Template." The dialog is divided into several sections:

- Identification:** Rule Name: Socks Monitor Traffic; Description: Inbound Socks Monitor Traffic; Action: Permit; Protocol: tcp.
- Source Port / ICMP Type:** Operation: Greater than; Port #/Type: 1023.
- Destination Port/ICMP Code:** Operation: Equal to; Port #/Code: 5051.
- Interfaces Settings:** Interface: Secure.
- Direction/Control:** Routing: local; Direction: inbound; Log Control: No; Frag. Control: Yes.

Buttons at the bottom include OK, Cancel, and Help.

Figure 452. Incoming SOCKS Monitor Traffic Rule

The next rule is for the outgoing traffic.

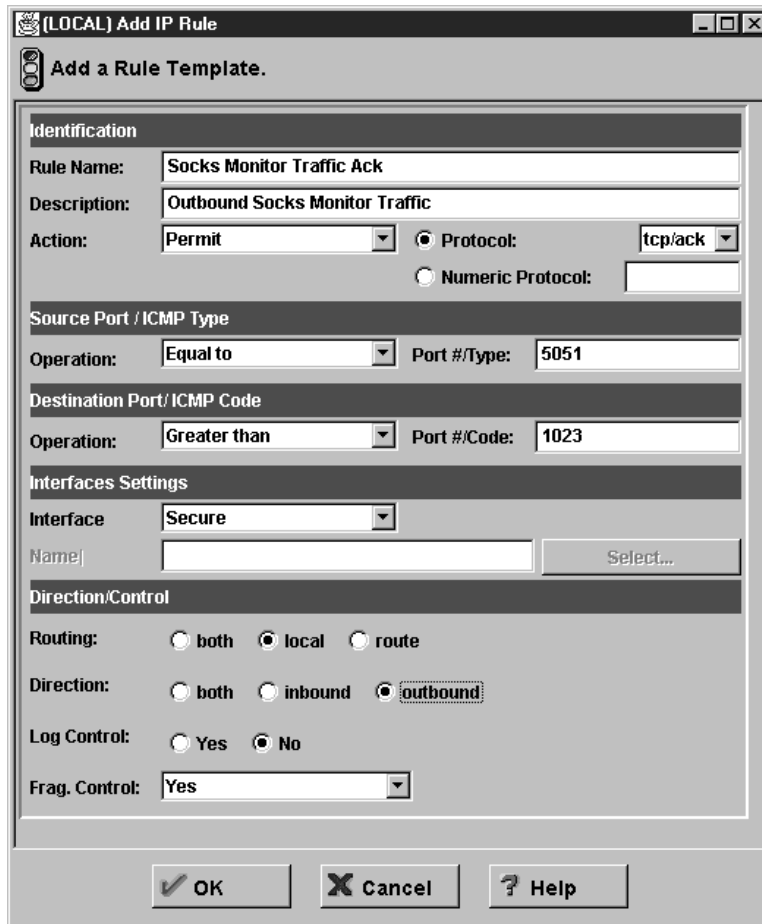


Figure 453. Outgoing SOCKS Monitor Traffic Rule

Now we create a service from both rules. Be careful with the direction of flow.

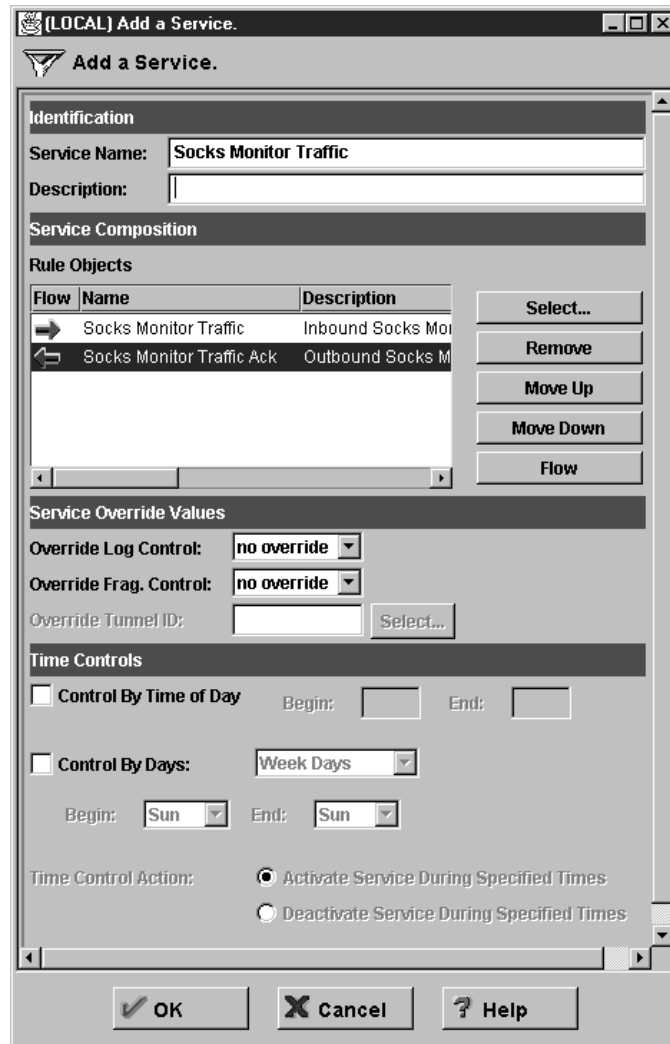


Figure 454. SOCKS Monitor Traffic Service

Finally we can create the connection using the new service.

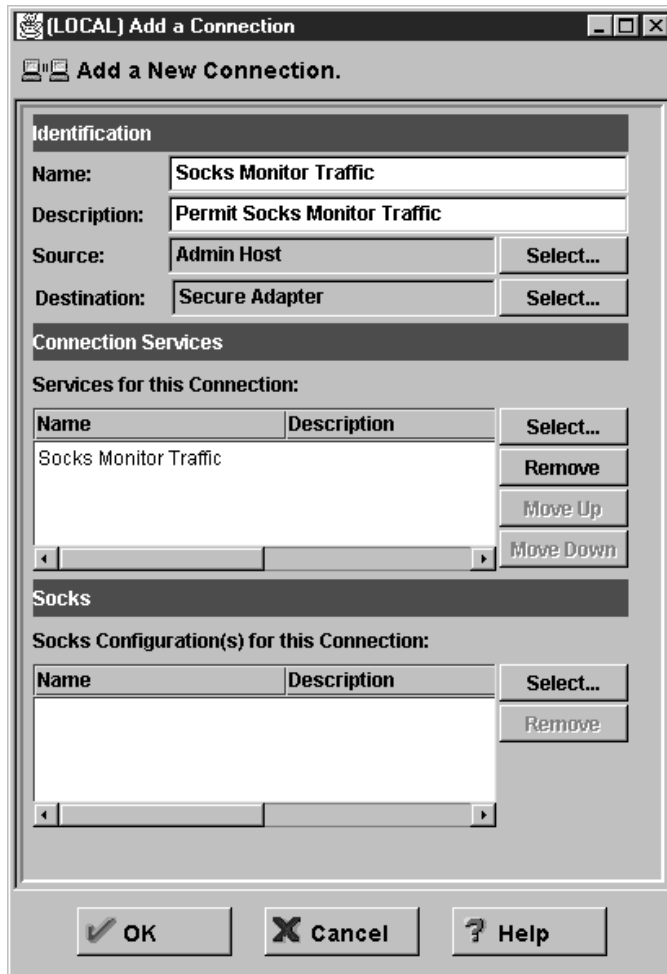


Figure 455. SOCKS Monitor Traffic Connection

Don't forget to regenerate the connection rules to activate the new connection.

16.5.2 Starting the SOCKS Traffic Monitor

To start the application select **Start** → **Programs** from the Windows NT task bar. Select either **IBM Firewall** (on the firewall itself) or **IBM Firewall (Client)** (on a remote machine). From the next menu select **SOCKS Monitor**. You now have an empty SOCKS Traffic Monitor window.

16.5.3 Monitor Traffic

When you start the Traffic Monitor you get only an empty window. You need first to create a new monitor by selecting **SOCKS5** from the menu bar. From the pop-up menu select **New**. Your window should now look like Figure 456 on page 445.

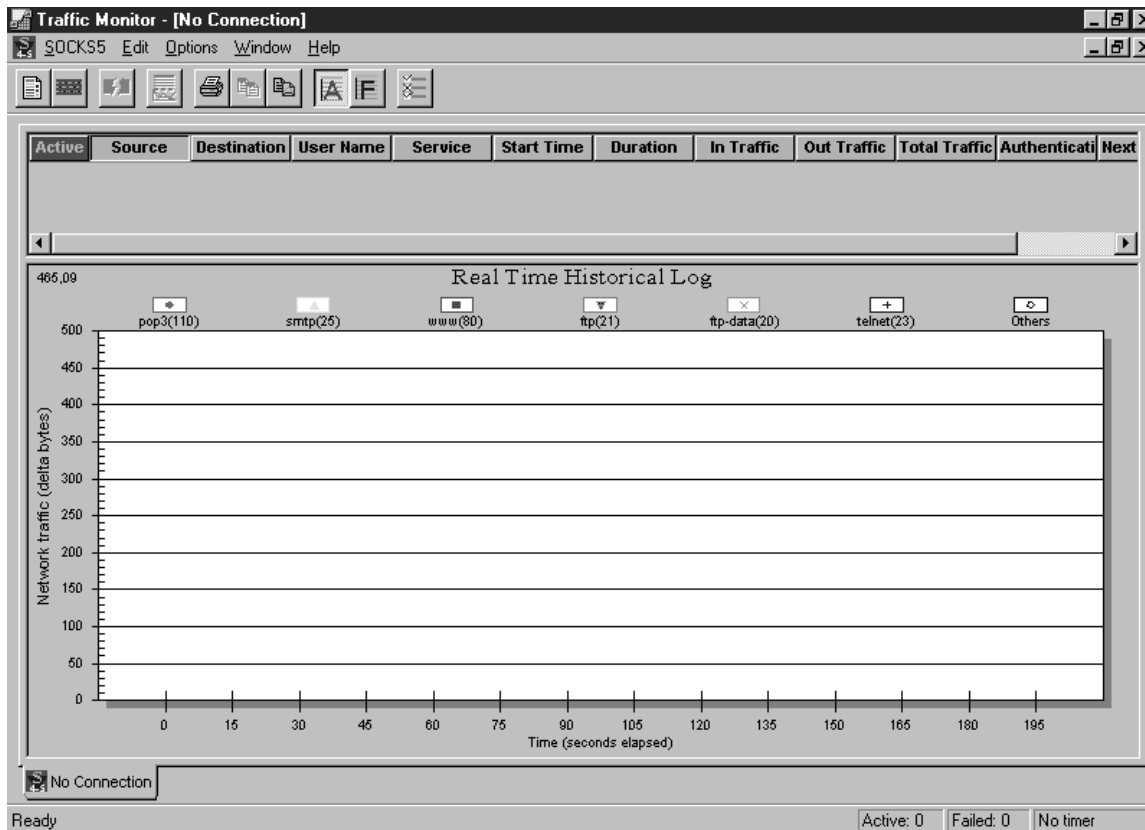


Figure 456. SOCKS Traffic Monitor without Connection

From this window you can connect to any IBM Firewall with a SOCKS 5 server. To do so, select **SOCKS5** from the menu bar and **Connect** from the following pull-down menu. The application asks you for the address of the SOCKS V5 server (see Figure 457).



Figure 457. Connection to SOCKS V5 Server

Figure 458 on page 446 shows the window you get when you enter the address and click the **OK** button. Please remember that the IBM Firewall SOCKS Monitor Service must run on the firewall and you must have a valid IP filter to allow the traffic to connect to the SOCKS V5 server.

Figure 458 on page 446 shows a sample output from the monitor.

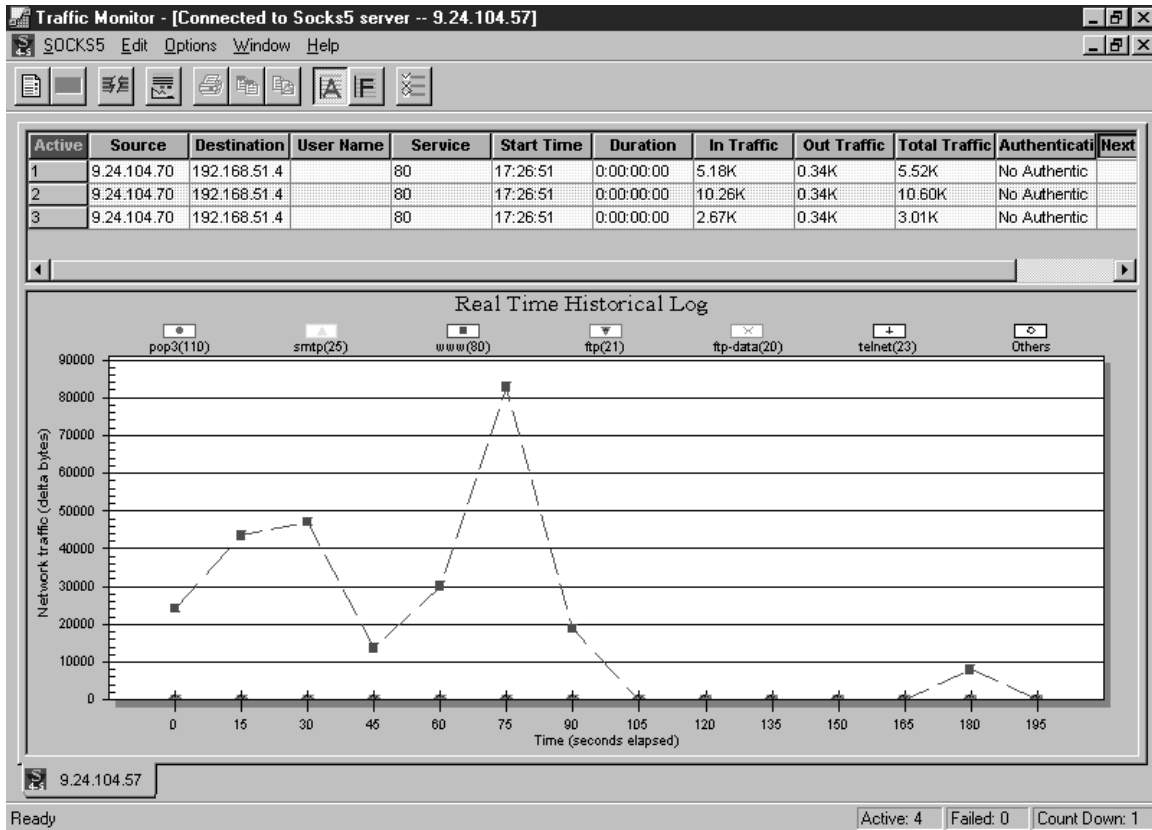


Figure 458. Sample Traffic Monitor Output

The window is divided into two parts. The upper part shows you the current active sessions to the SOCKS V5 server with detailed data about the source and destination address, type of service, and time, size, and authentication information.

The lower part shows you real-time historic log of the activity on the SOCKS V5 server. You see separate lines for every service monitored. The following services are monitored by default.

Type of Service	Port
ftp	21
ftp-data	20
pop3	110
smtp	25
telnet	23
www	80
other	

You can also edit the monitored services.

16.5.4 Edit Traffic Monitor

To edit the properties of the SOCKS Traffic Monitor select the **Options** entry from the Options menu. You get a window with three tabs:

- Table
- Data
- Graph

Clicking on the **Table** tab lets you edit the columns shown in the active connections table in the upper part of the main window.

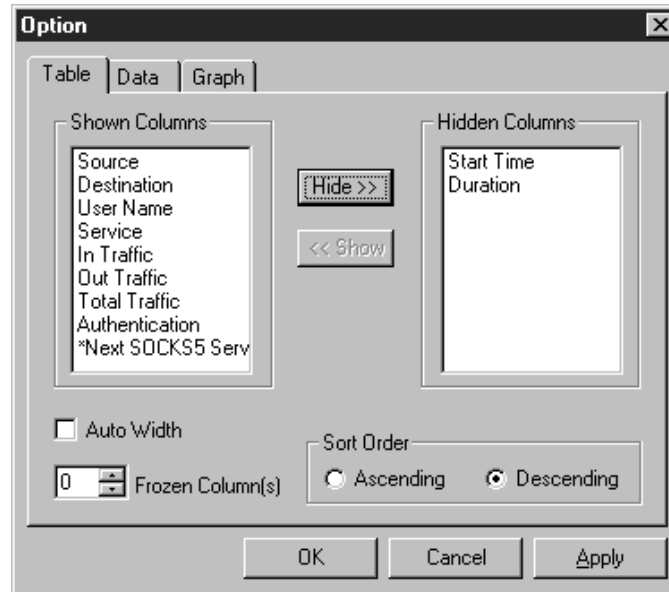


Figure 459. SOCKS Monitor Table Settings

Clicking on the **Data** tab lets you customize the data format shown in the active data connections table.

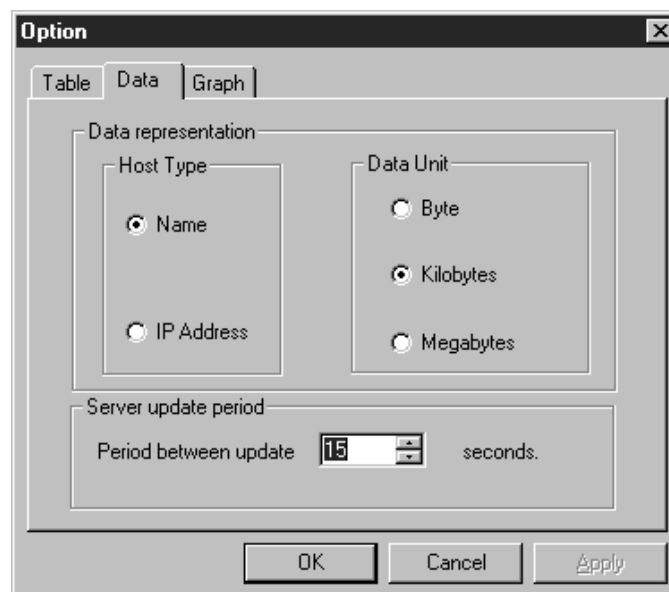


Figure 460. SOCKS Monitor Data Settings

Clicking on the **Graph** tab enables you to edit the services that should be monitored. You can add new ones, edit and delete existing ones.

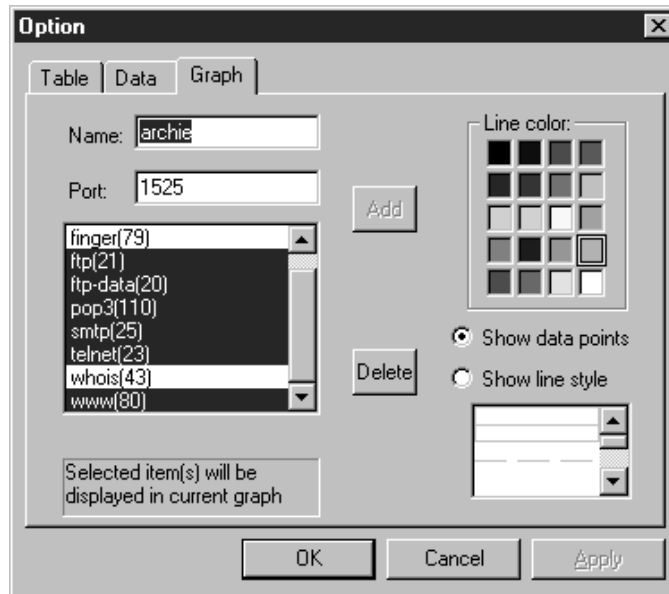


Figure 461. SOCKS Monitor Graph Settings

You also can define and change the line type and the color used for that kind of traffic in the monitor window.

Chapter 17. Configuring the IBM Firewall for Three Adapters

The utilization of three adapters in the IBM Firewall is an alternative for a configuration using two IBM Firewalls, with a DMZ (demilitarized zone) between them as shown on Figure 462 or an IBM Firewall plus a router.

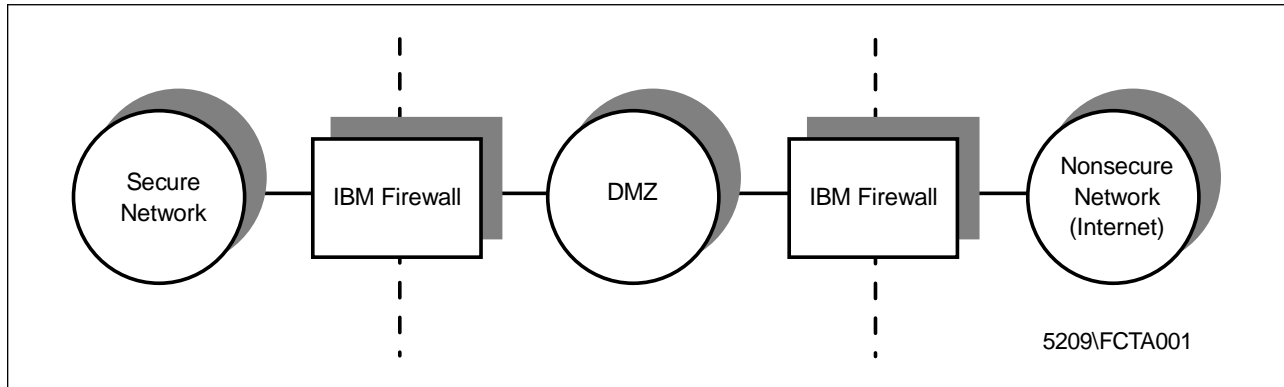


Figure 462. DMZ with Two Firewalls

By using a third adapter in the IBM Firewall, you have a mid-term solution between a cheaper configuration with one firewall and a more expensive configuration with two firewalls. The architecture of both configurations is similar, but the level of security of the two IBM Firewalls configuration is higher by providing a second barrier in case the external firewall is compromised. Generally the three adapters configuration is used when you need some interaction with systems in the secure network from a non-secure network (for example, the Internet). As any communication from the non-secure network directly to the secure network has to be avoided, this configuration allows you to have servers in the DMZ with a lower exposure level than the public Web server that's in the non-secure network without compromising the security of the internal network as shown in Figure 463.

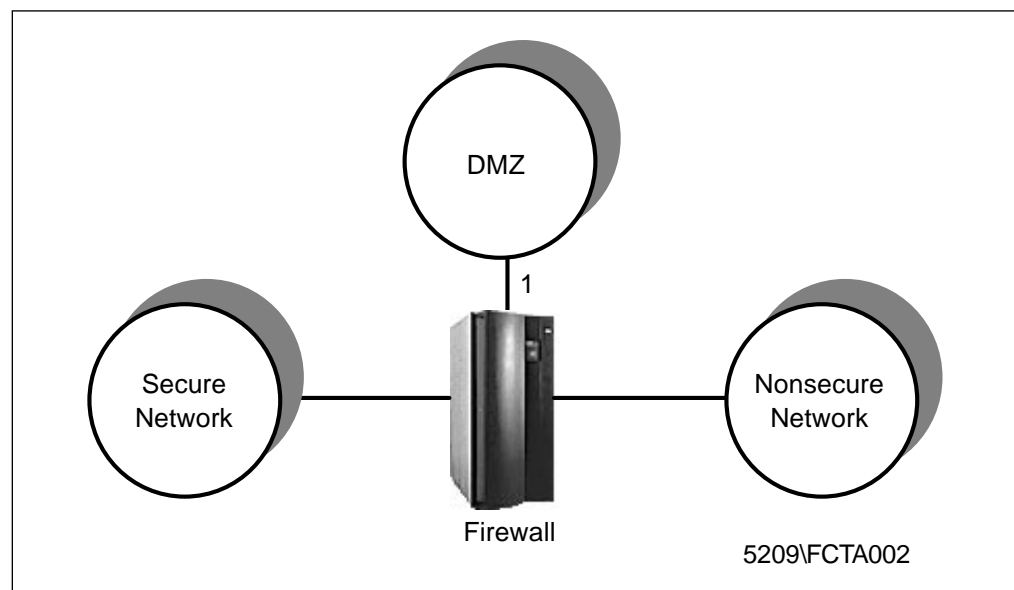


Figure 463. Three Adapters Configuration

17.1 Considerations with Two Non-secure Interfaces

The IBM Firewall sets up the additional interface as a non-secure interface. In this case we have one secure interface to the internal network and two non-secure interfaces, one to the Internet and another to the DMZ. We will have to add some specific filter rules to our firewall to assure that the addition of the third interface will not put our internal network at risk of an external attack. See Chapter 9, "Examples of Rules for Specific Services" on page 161 for more information on how to configure rules for specific services. We will have to use an anti-spoofing rule to avoid an attacker impersonating the IP addresses of the servers in the DMZ to access the internal network (see Figure 464).

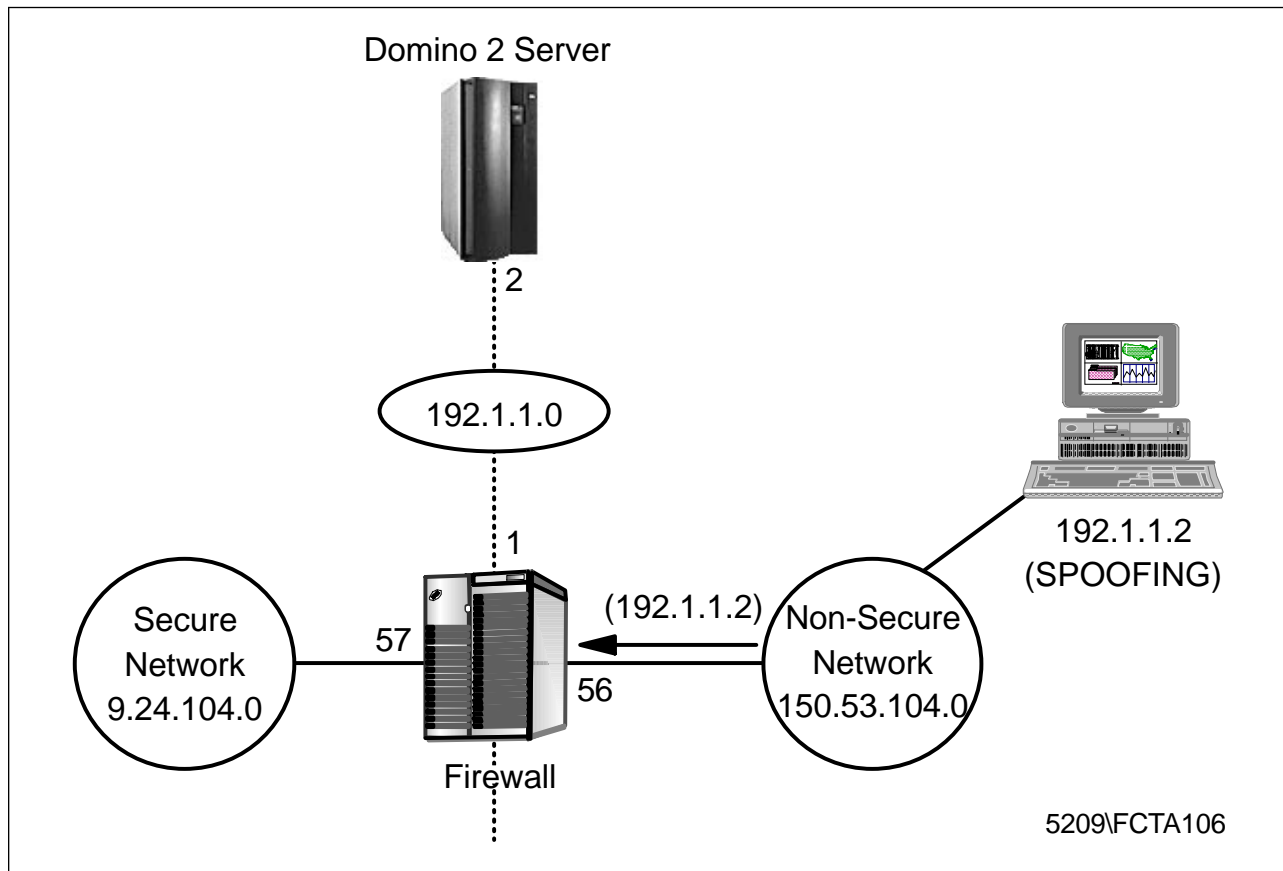


Figure 464. IP Spoofing

The predefined anti-spoofing rule can only deny inbound traffic at all non-secure interfaces. Since we have two non-secure interfaces, if we use the predefined anti-spoofing rule, we will be denying inbound traffic on both interfaces. In this way, servers in the DMZ will not be able to communicate to the internal network. To solve this problem we name each non-secure interface. For example, DMZ for the interface connected to the DMZ and Internet to the interface connected to the Internet. Doing that, the IBM Firewall will be able to deny traffic exactly in which interface we want.

17.2 Configuration Example

In this section we describe a solution using two Domino servers that permit database access to Clients (Notes clients or browsers) from the Internet as shown in Figure 465.

In Chapter 8, "IBM Firewall Rule Base" on page 143 we explain how to set up rules, services and connections. Here we only show one example of using a third adapter.

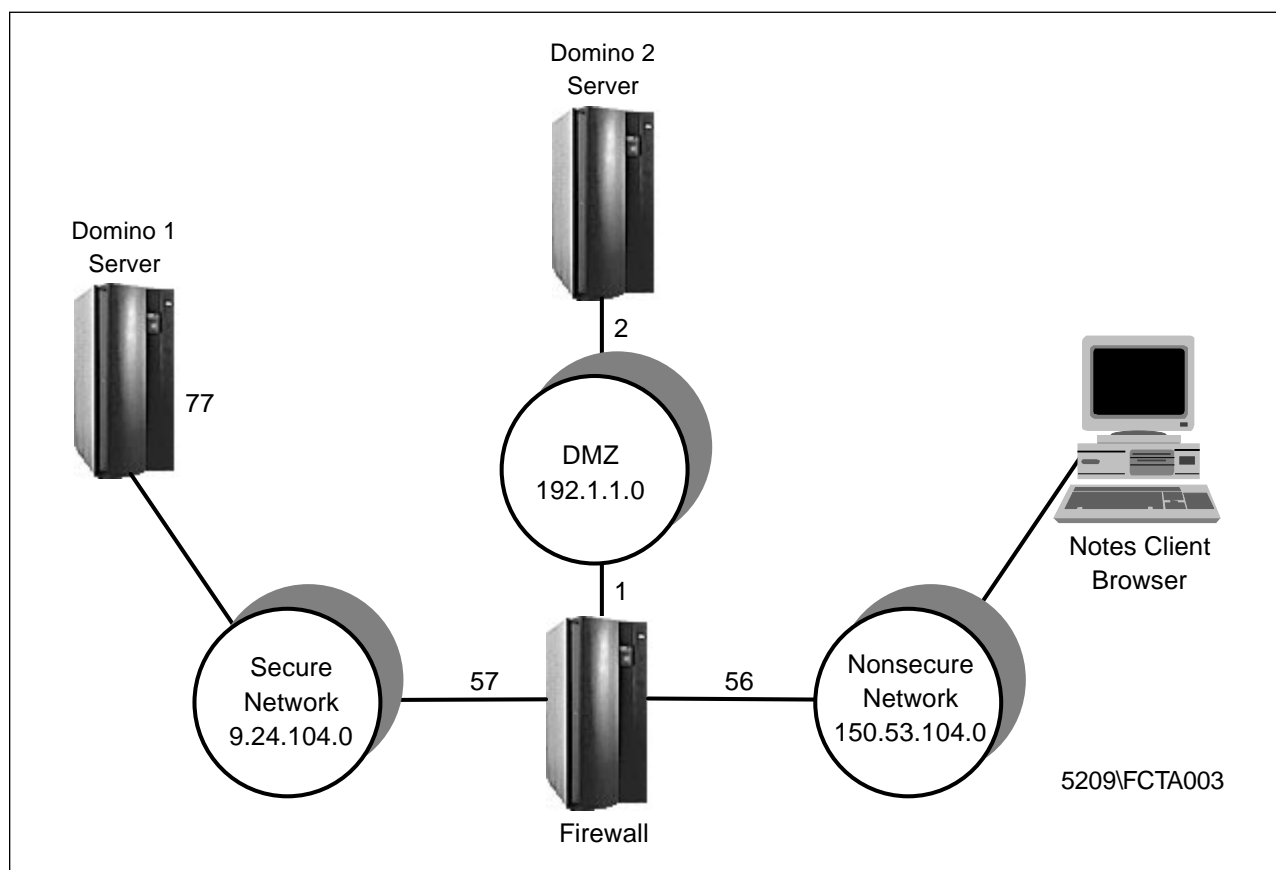


Figure 465. Three Adapters Example

One server is located in the secure network (Domino1) and the other one in the DMZ (Domino2). Clients will not have direct access to the Domino1 server. The Domino1 server will replicate its databases to the Domino2 server. Clients will access the Domino2 server in the DMZ using SSL (Secure Socket Layer) which will be the only protocol allowed to pass from the non-secure network to the DMZ.

The connection configured between the Domino1 and the Domino2 servers has to permit traffic only between these two servers: TCP packets (TCP as defined in the IBM Firewall rules) from the Domino1 server to the Domino2 server and TCP/ACK packets from the Domino2 server to the Domino1 server.

This rule will minimize the risk of IP spoofing attacks. No other machine outside the secure network should be permitted to communicate with the Domino1 server; only Domino2 is allowed.

17.2.1 Installing the Third Adapter

In this section we describe the procedure to install the additional adapter in the IBM Firewall machine. In our case, we could not use more than two token-ring adapters at the same machine due to an I/O address limitation of the hardware; so we had to install an Ethernet adapter as our third adapter. You may experience some interrupt problems when installing the additional adapter. If all the interrupts of the machine are in use, try to disable either the serial or the parallel port. After that you will have to configure the adapter to use an available interrupt.

To add an adapter:

1. From the NT Desktop, select **Start** → **Settings**→**Control Panel**.
2. Select the **Network** icon.
3. From the Network dialog, select the **Adapters** tab.
4. Choose **Add** to see the list of supported adapters. If the name of your adapter is not on the list, choose **Have Disk** and follow the prompts.
5. Insert the NT disk into the CD-ROM drive or specify a location where NT can find the support files necessary to complete installation.
6. After the files have been copied and the adapter installed, close the Network dialog and reboot the machine.

To add the IP protocol:

1. From the NT Desktop, select **Start** → **Settings**→**Control Panel**.
2. Select the **Network** icon.
3. From the Network dialog select the **Protocols** tab.
4. Choose **Add** to see the list of supported protocols.
5. Choose **TCP/IP Protocol**.
6. Click on **OK**.
7. Insert the NT Disk into the CD-ROM drive.
8. Close the Network dialog and reboot the machine.
9. After rebooting the machine, repeat steps 1-5 and select **Details** to enter the TCP/IP parameters of the adapter (Hostname, IP address, mask, default gateway, DNS server, etc)
10. Click on **OK**.
11. Close the Network dialog and reboot the machine.

The IBM Firewall will set up the additional interface (adapter) as a non-secure interface.

After installing the third adapter, run the Configuration Client and select **System Administration** → **Interfaces** to see the list of installed interfaces.

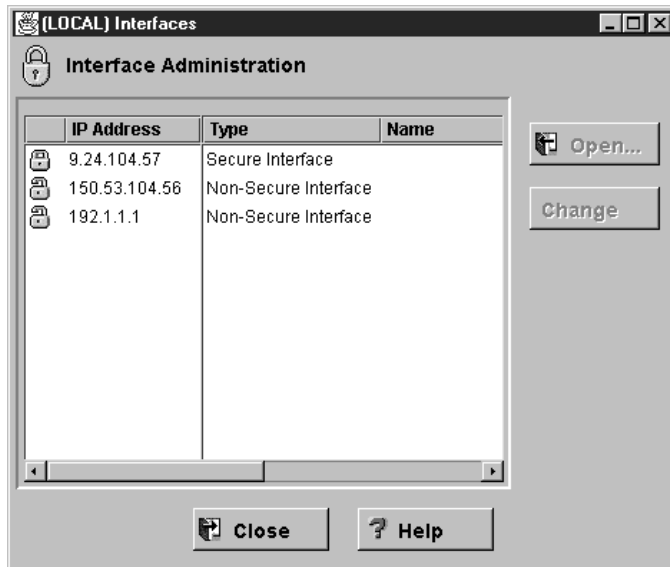


Figure 466. Installed Adapters

Once we have two non-secure adapters and have to configure traffic between them, we should give them a name to identify to which non-secure adapter the data is to be sent or received. Select the new adapter from the list of interfaces and click on **Open**. Enter a name for the interface (in our example we use DMZ) and click on **OK**.

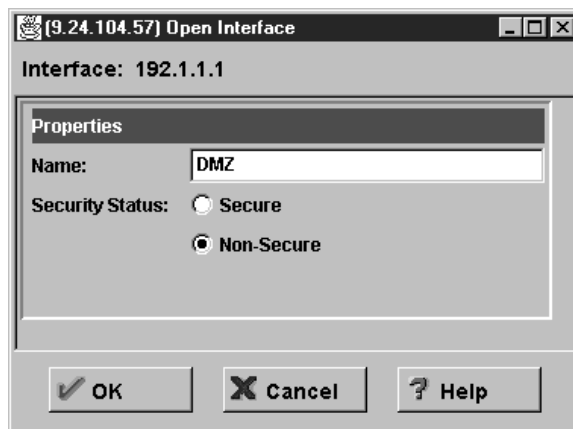


Figure 467. Naming an Interface

Repeat the previous steps to name the other non-secure interface. We use the name *Internet* for Internet interface. Your Interfaces window now appears (see Figure 468 on page 454).

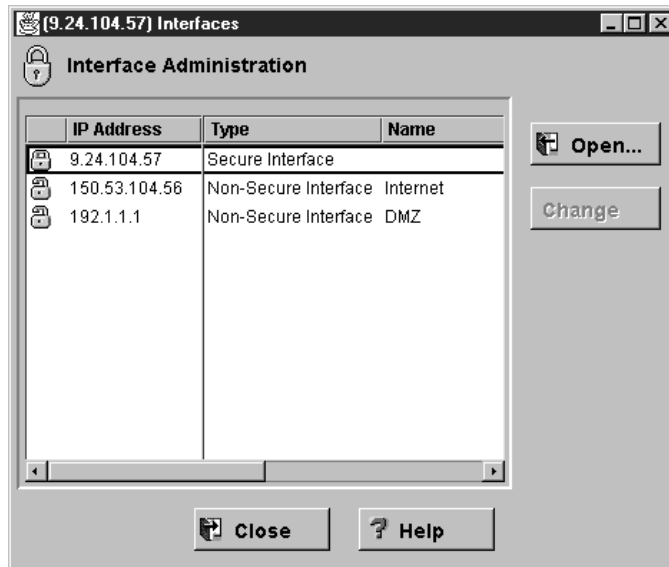


Figure 468. Interfaces Window

17.2.2 Adding Network Objects

Now that we have added the third adapter, we need to create the network objects for the third interface, the Domino server in the DMZ and the Domino server in the secure network.

To add the objects:

1. From the Configuration Client navigation tree select **Network Objects**.

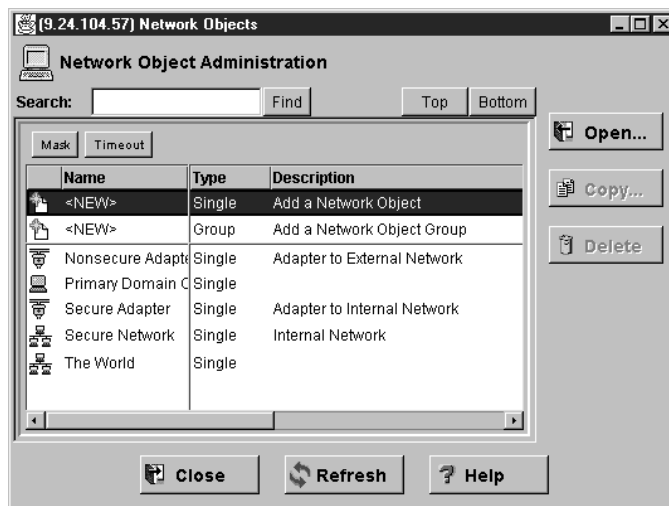


Figure 469. Adding Network Objects

2. Select **<NEW>** with Type Single and click on **Open**.

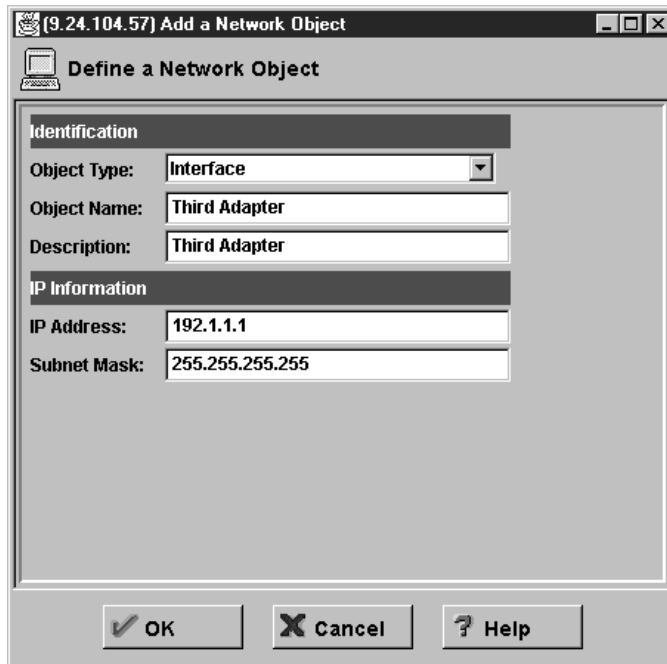


Figure 470. Third Adapter Object

3. Enter the data related to the third adapter as shown in Figure 470.
4. Click on **OK**.

Repeat the previous steps for servers Domino1 and Domino2 but don't forget that the object type for the servers is *Host*.

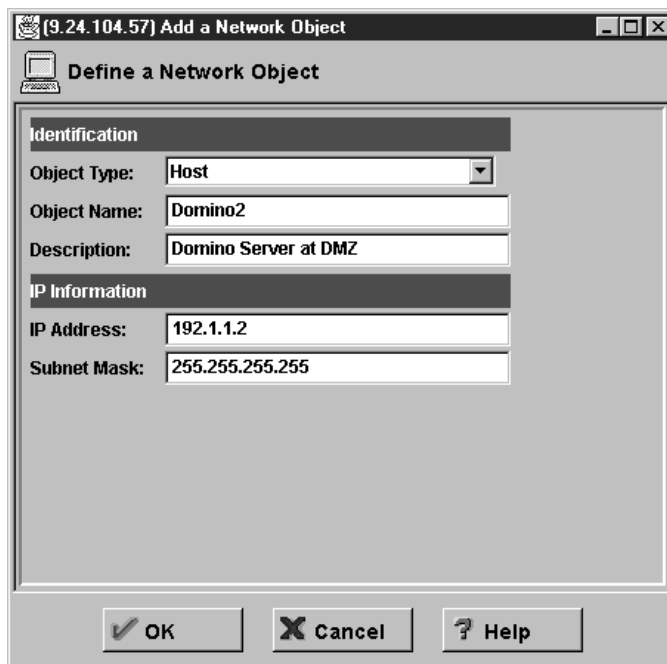


Figure 471. Adding Domino2 Object

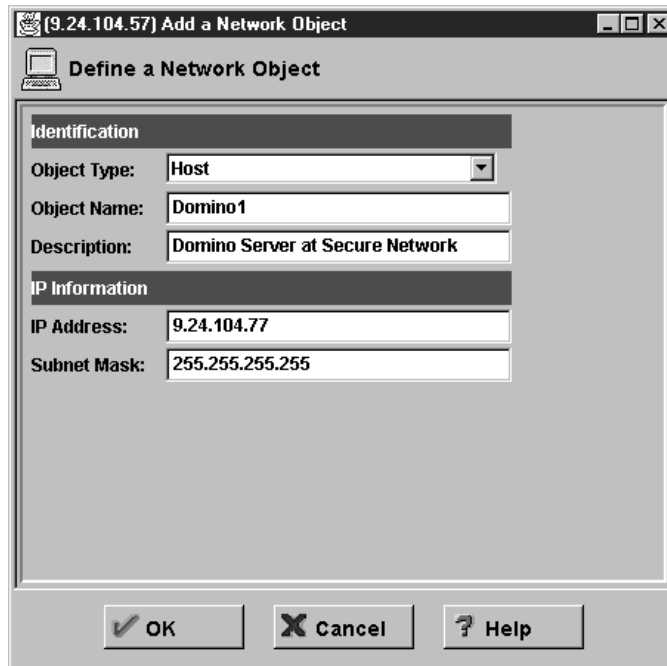


Figure 472. Adding Domino1 Object

17.2.3 Configuring Connection from Internet to Domino2 Server

Predefined rules may be used for connections whenever it is possible. If we don't have predefined rules that can be used in our configuration, we will have to set up our own rules and services for each connection, or copy data from predefined rules and services. Be aware that the more complicated environment you have, the easier it is to get security holes. So pay attention on each rule you create and check them thoroughly before enabling them. See Chapter 8, "IBM Firewall Rule Base" on page 143 for more information about configuring rules and services.

We will create the rules, the service and the connection between the Internet and the Domino2 that is in the DMZ.

Select **Traffic Control** → **Connection Templates** → **Rules** from the Configuration Client navigation tree.

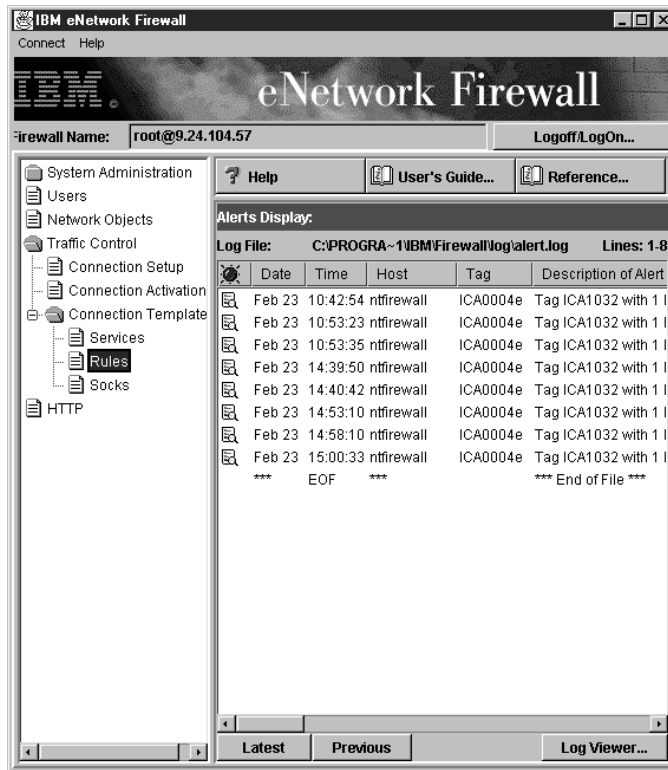


Figure 473. Configuration Client Navigation Tree

We will copy some predefined rules and make the changes we need to create our own rules. Figure 474 shows the rules we have to create.

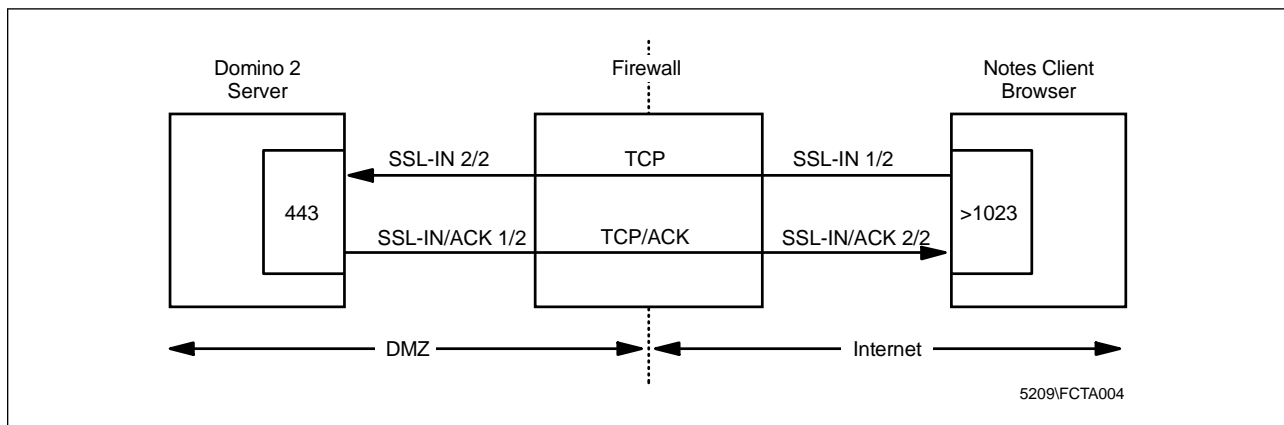


Figure 474. Rules Configuration

In this case we will use the HTTPS rules because they are very similar to the rules we have to create, requiring minor changes.

Select the **HTTPS 1/2** rule from the rule list window and click on **Copy**.

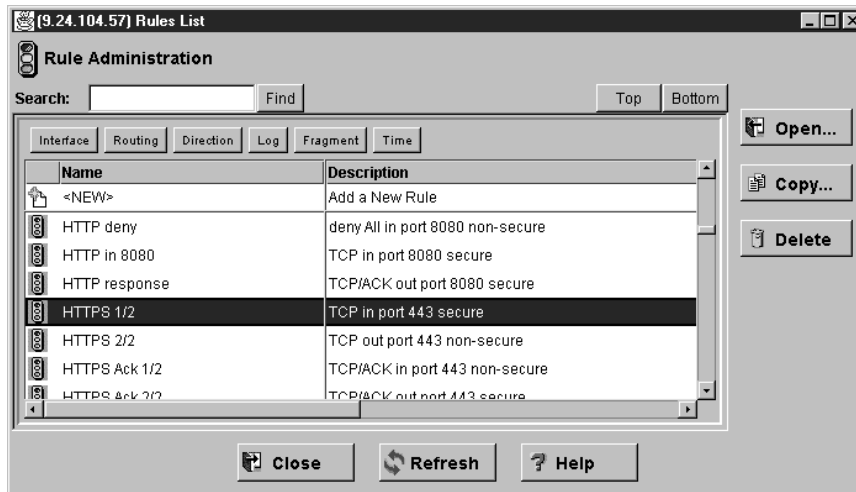


Figure 475. Rule List Window

We only have to change the interface from secure to Internet. Our first rule will be called SSL-IN 1/2.

1. Enter the rule name.
2. Select **Specific** for the interface.
3. Click on **Select** to see the list of interfaces.
4. Select the interface with the name **Internet**.
5. Click on **OK**.
6. Select **Yes** for the log control, so that every packet that matches that rule will be logged.

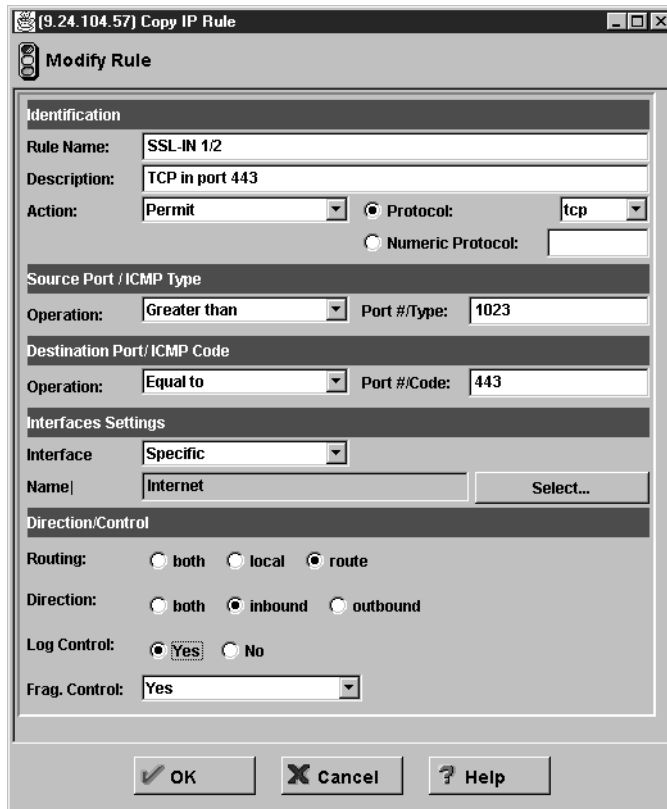


Figure 476. Creating Rule SSL-IN 1/2

Repeat the same procedures for rules HTTPS 2/2, HTTPS Ack 1/2 and HTTPS Ack 2/2 to create the rules SSL-IN 2/2, SSL-IN/ACK 1/2 and SSL-IN/ACK 2/2. Enter the data described in the following pictures.

For rule SSL-IN 2/2 see Figure 477 on page 460. We changed the interface from *non-secure* to *DMZ*.

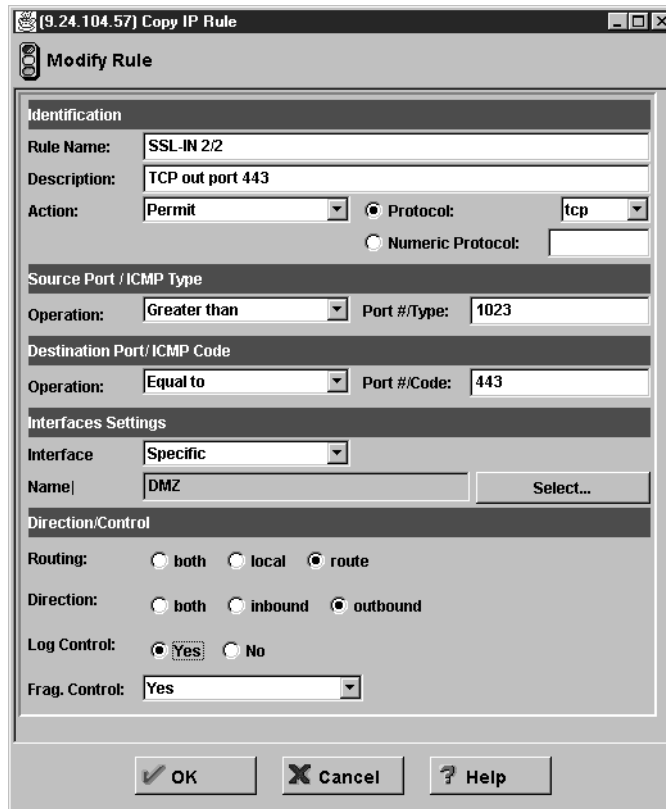


Figure 477. Creating Rule SSL-IN 2/2

For rule SSL-IN/ACK 1/2 see Figure 478 on page 461. We changed the interface from *non-secure* to *DMZ*.

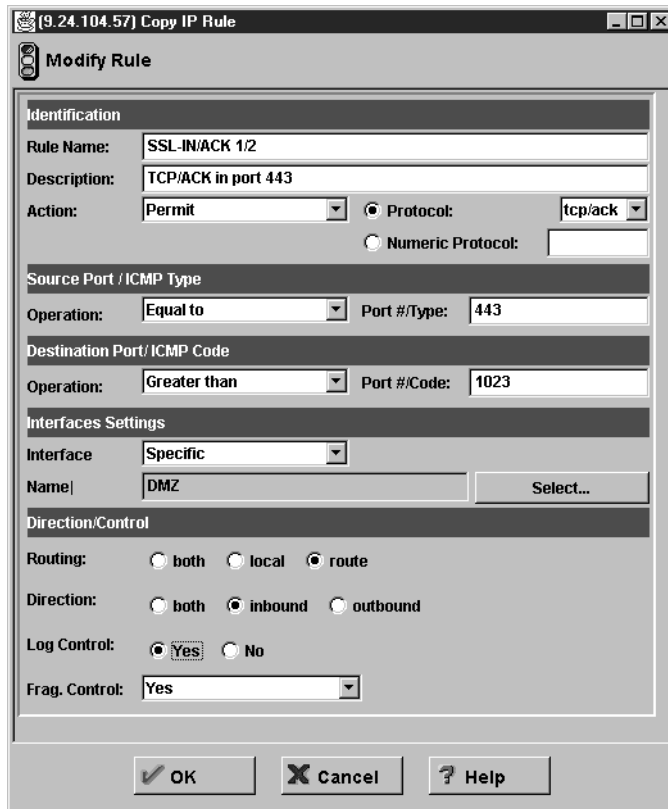


Figure 478. Creating Rule SSL-IN/ACK 1/2

For rule SSL-IN/ACK 2/2 see Figure 479 on page 462. We changed the interface from secure to Internet.

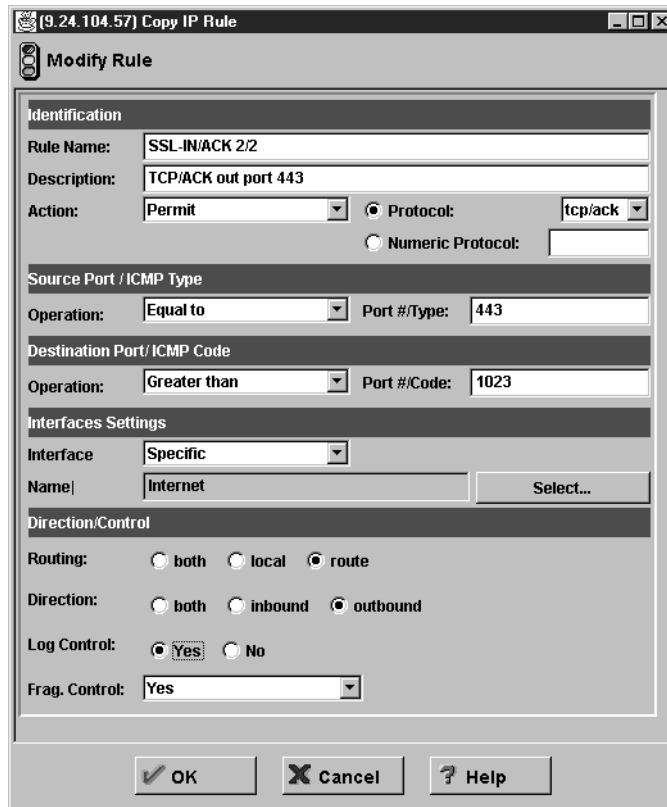


Figure 479. Creating Rule SSL-IN/ACK 2/2

After completing the SSL-IN/ACK 2/2 rule and clicking **OK**, check if the rules you have just created appear in the Rules List window (Figure 480).

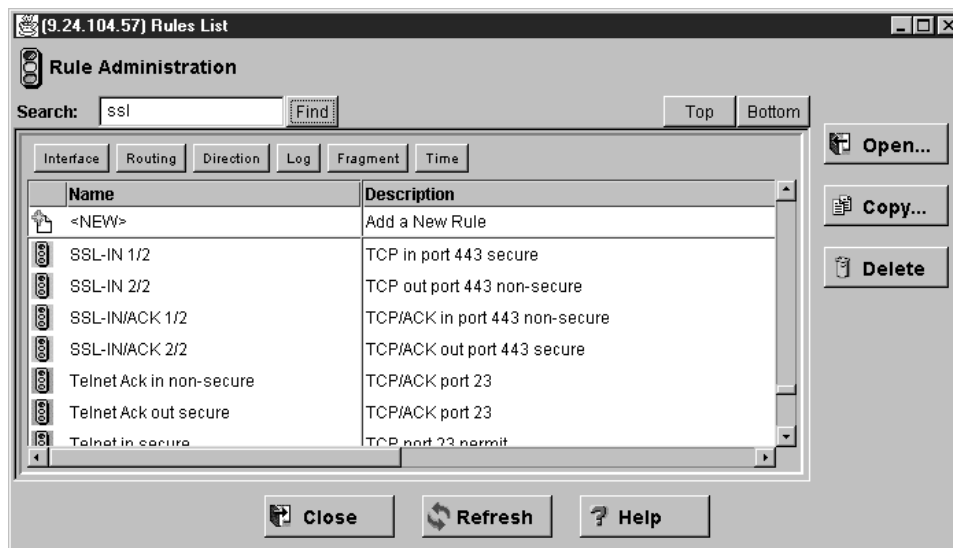


Figure 480. New SSL Rules

Now that we have created the rules, we have to create a service that will use the new rules.

1. At the Configuration Client navigation tree select **Services**.
2. At the Services List dialog select **<NEW>** and click on **Open**.

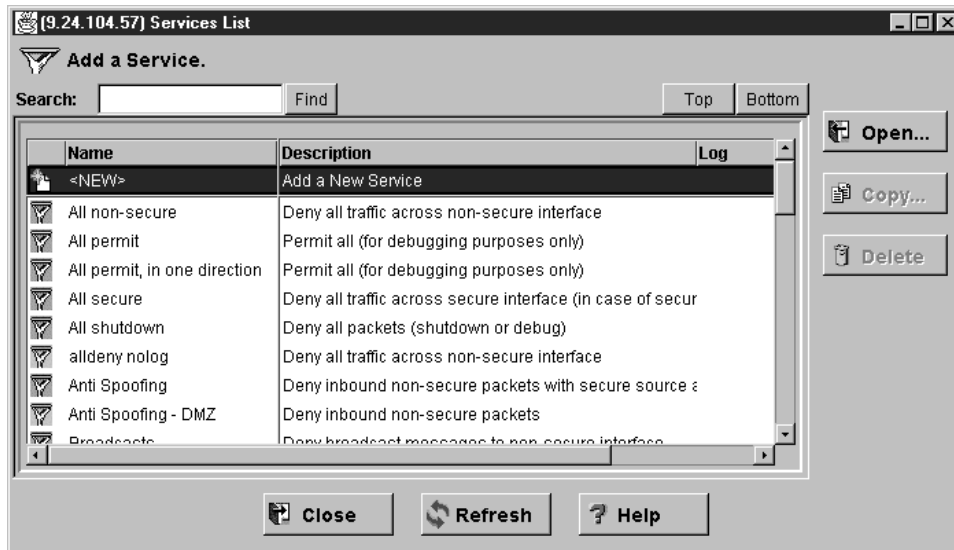


Figure 481. Services List

3. At the Add a Service dialog enter the name of the service and a description to identify the service. In our case the name of the service will be SSL-IN.
4. Click on **Select** to view the rules list and to select the set of rules that will compose the service.
5. Highlight the rules by holding down the control key and clicking on the rules you want to add; then click on **OK**.

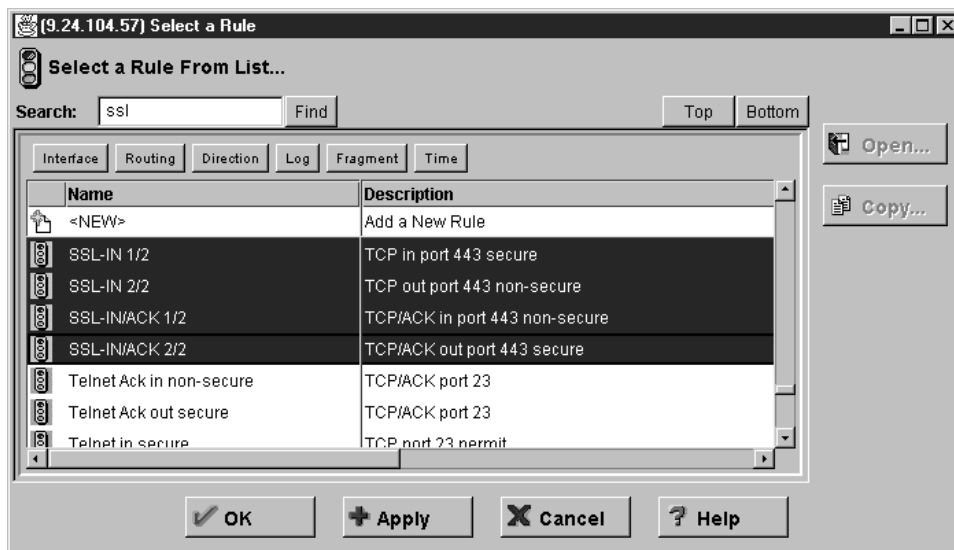


Figure 482. Rules Selection

6. At the Add a Service window we will have all the SSL-IN rules with a green arrow at the left side (pointing to the right). A green arrow indicates the rule applies to packets flowing from the source to the destination object. Since the SSL-IN/ACK rules apply to packets flowing from the destination to the source (see Figure 474 on page 457), you will have to change the flow direction of these rules. To change the flow direction select the rule **SSL-IN/ACK 1/2** and click on **Flow**. See Chapter 8, "IBM Firewall Rule Base" on page 143 for more

information on flow control. Do the same for the rule SSL-IN/ACK 2/2. The green arrows will change to blue ones (pointing to the left).

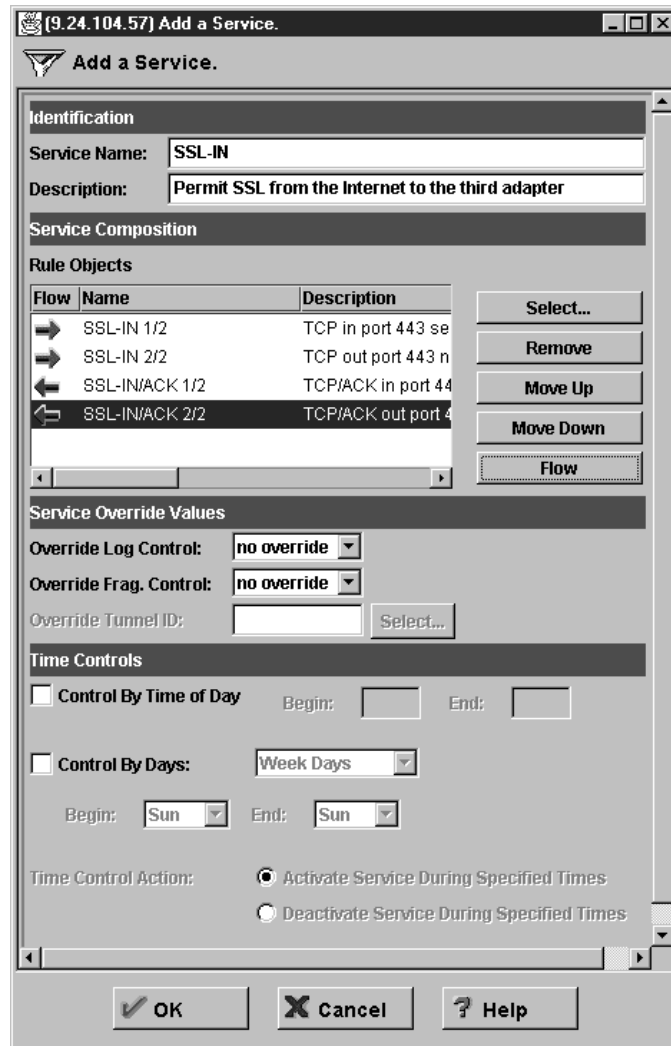


Figure 483. Changing Flow

7. Now click on **OK**.

Since we have the service configured; we can now create the connection between the Internet and the Domino2 server.

1. From the Configuration Client navigation tree select **Connection Setup** → **<NEW>** → **Open**.
2. Enter the name and a description of the connection.
3. For the source field click **Select** and choose **The World** object.
4. For the destination field click **Select** and choose the **Domino2** object.
5. For the connections services click **Select** and choose the **SSL-IN** service from the list and click **OK**.

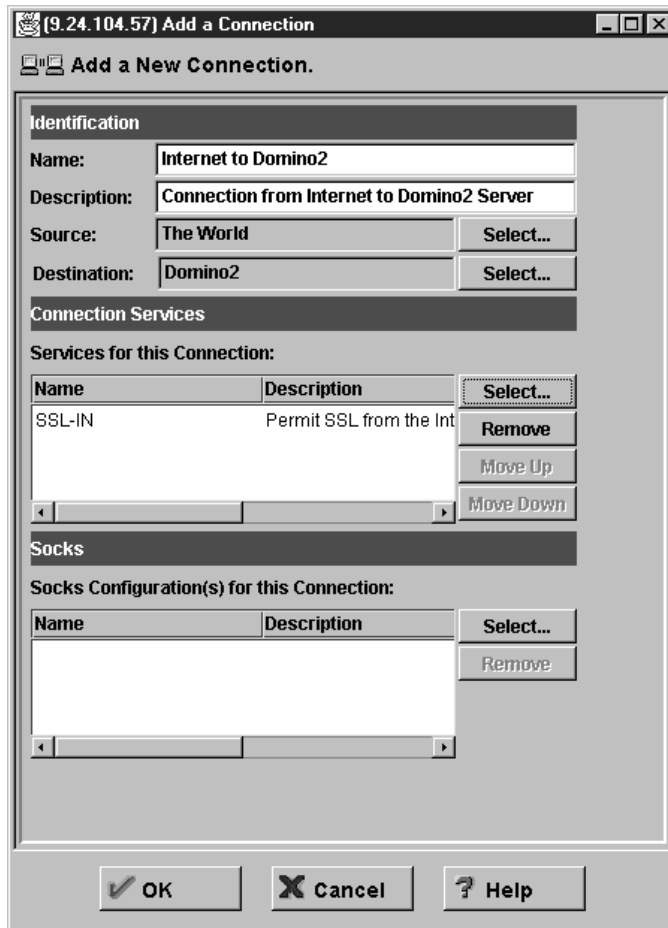


Figure 484. Internet to Domino2 Connection

6. Select **OK**.

```
#Between The World and Domino2
#      Service : SSL-IN
# Description : Permit SSL from the Internet to the third adapter
permit 0 0 192.1.1.2 255.255.255.255 tcp gt 1023 eq 443 specific(Internet) route inbound l=y f=y
permit 0 0 192.1.1.2 255.255.255.255 tcp gt 1023 eq 443 specific(DMZ) route outbound l=y f=y
permit 192.1.1.2 255.255.255.255 0 0 tcp/ack eq 443 gt 1023 specific(DMZ) route inbound l=y f=y
permit 192.1.1.2 255.255.255.255 0 0 tcp/ack eq 443 gt 1023 specific(Internet) route outbound l=y f=y
```

Figure 485. Internet to Domino2 Connection Rules

In Figure 485 we can see all the rules that apply to the connection between the Internet and the Domino2 Server.

17.2.4 Connection from Domino1 Server to Domino2 Server

We will now create the rules, service and the connection between the Domino1 server and the Domino2 server. Since we don't have any similar predefined rules to copy data, we will create all of the rules. See 9.13, "Lotus Notes and Domino" on page 199 for further information. Figure 486 on page 466 shows the rules we have to create.

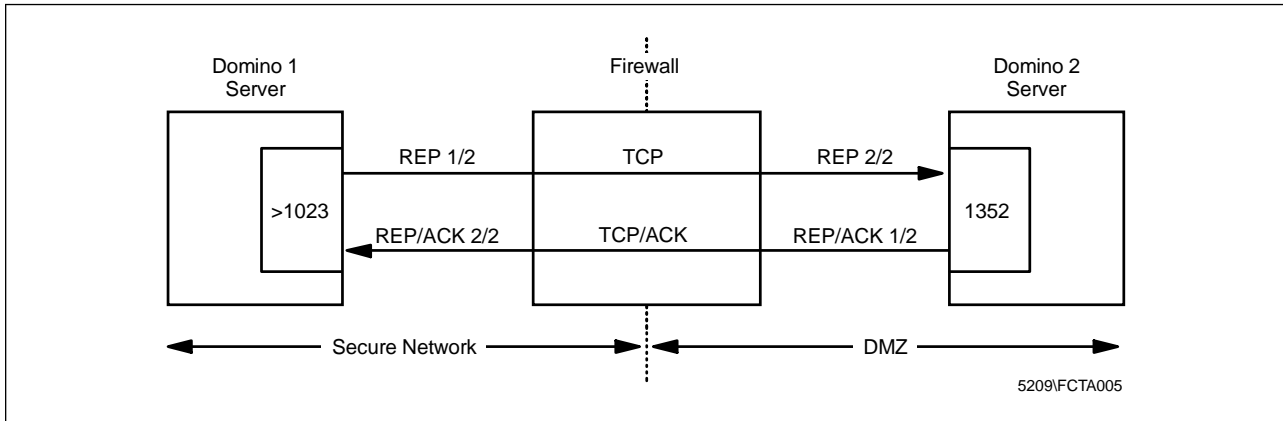


Figure 486. Rules Configuration

Select **Traffic Control** → **Connection Templates** → **Rules** at the Configuration Client navigation tree window.

Let's create the rules REP 1/2, REP 2/2, REP/ACK 1/2 and REP/ACK 2/2.

1. At the Rules List window select **<New>** and click on **Open**.
2. Enter the data as shown in Figure 487 for rule REP 1/2.

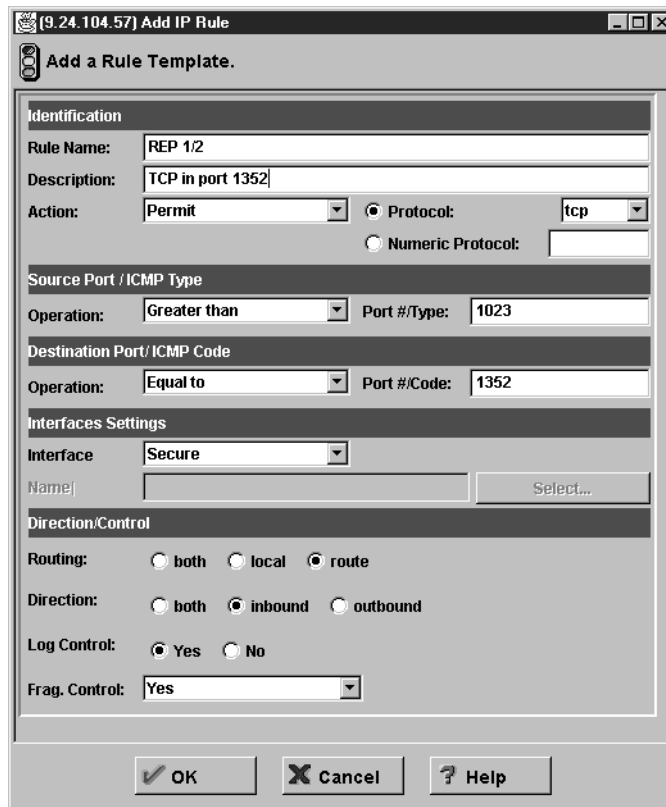


Figure 487. Adding REP 1/2 Rule

3. Click on **OK**.

Repeat the same procedures for rules REP 2/2, REP/ACK 1/2 and REP/ACK 2/2, entering the data described in the following pictures.

For rule REP 2/2 see Figure 488 on page 467.

(9.24.104.57) Add IP Rule

Add a Rule Template.

Identification

Rule Name: REP 2/2

Description: TCP out port 1352

Action: Permit Protocol: tcp

Source Port / ICMP Type

Operation: Greater than Port #/Type: 1023

Destination Port / ICMP Code

Operation: Equal to Port #/Code: 1352

Interfaces Settings

Interface: Specific

Name: DMZ Select...

Direction/Control

Routing: both local route

Direction: both inbound outbound

Log Control: Yes No

Frag. Control: Yes

OK Cancel Help

Figure 488. Creating Rule REP 2/2

For rule REP/ACK 1/2 see Figure 489 on page 468.

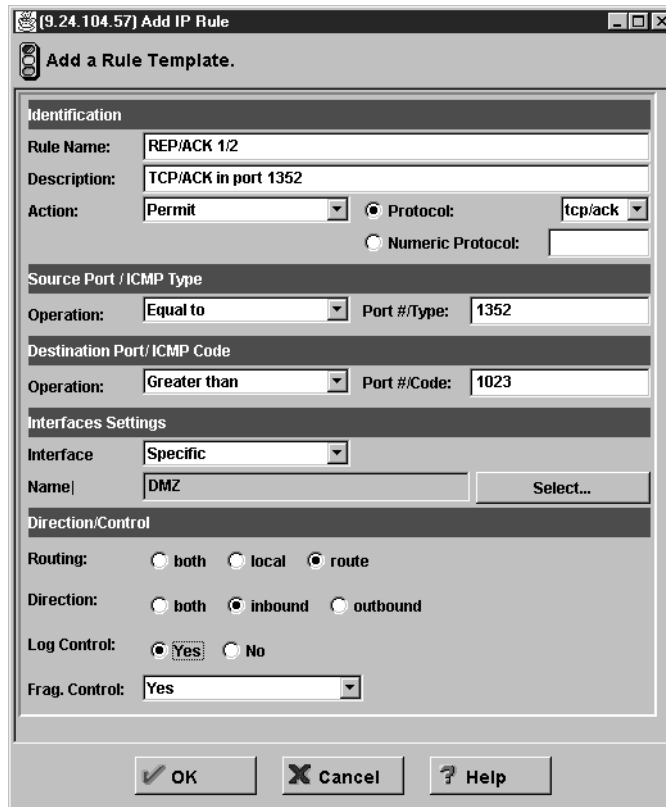


Figure 489. Creating Rule REP/ACK 1/2

For rule REP/ACK 2/2 see Figure 490 on page 469.

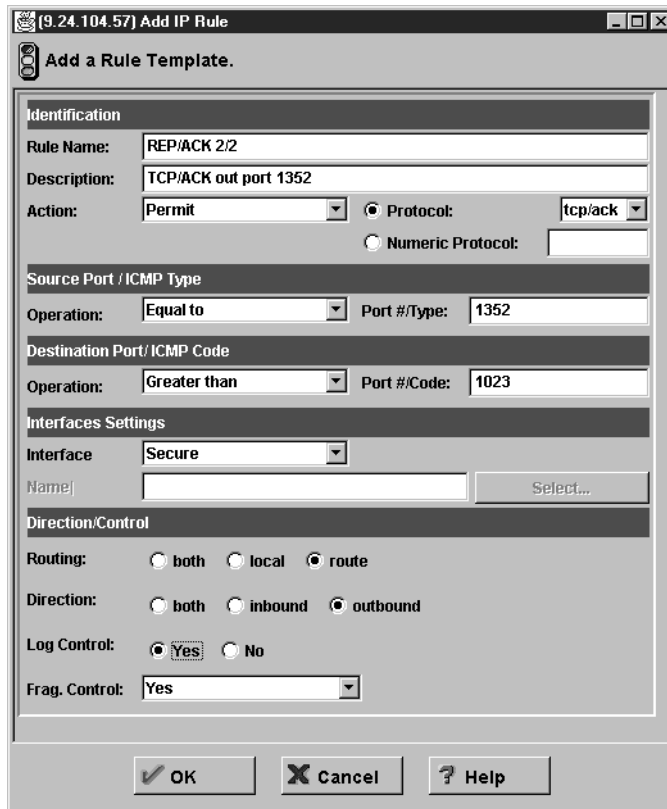


Figure 490. Creating Rule REP/ACK 2/2

After completing the REP/ACK 2/2 rule and clicking on **OK**, check if the rules you have just created appear in the Rules List window (Figure 491).

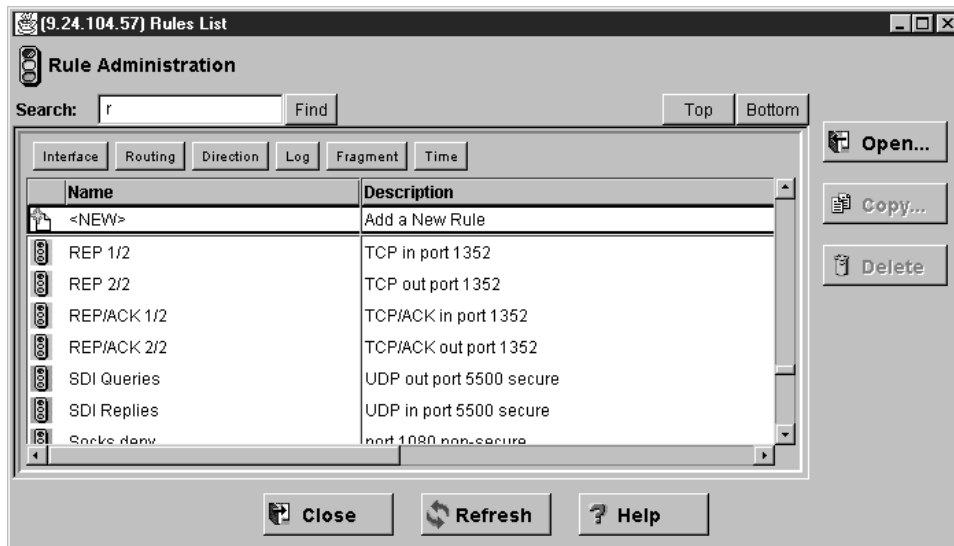


Figure 491. New REP Rules

Now that we have created the rules, we have to create a service that will use the new rules.

1. At the Configuration Client navigation tree select **Services**.

2. At the Services List dialog (Figure 481 on page 463) select **<NEW>** and click on **Open**.
3. At the Add a Service dialog enter the name of the service and a description to identify the service. In our case the name of the service will be *REPLICATION*.
4. Click on **Select** to view the Rule List and select the set of rules you need.

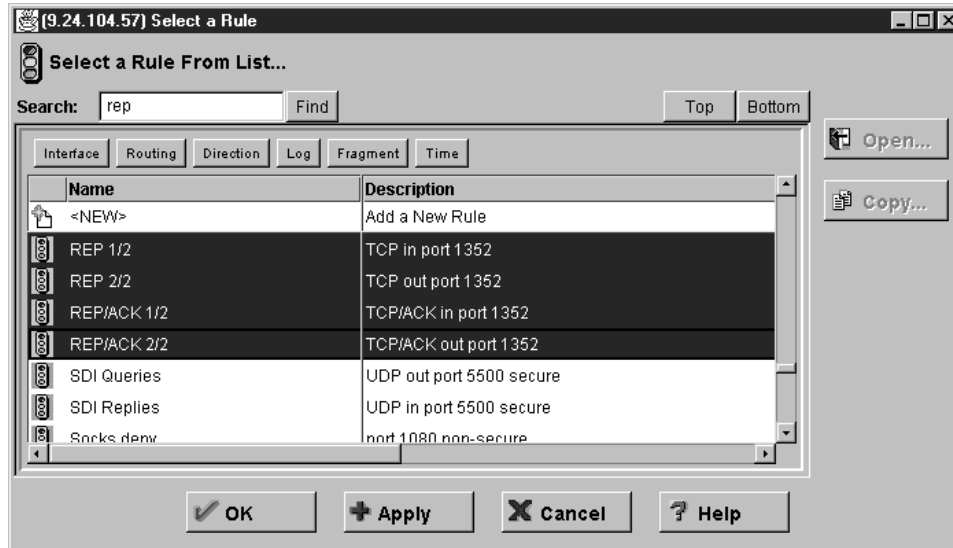


Figure 492. Rules Selection

5. Click on **OK**.
6. As in 17.2.4, "Connection from Domino1 Server to Domino2 Server" on page 465 you will have to change the flow control of the REP/ACK rules (see Figure 486 on page 466). At the Add a Service window select the rule **REP/ACK 1/2** and click on **Flow**. Do the same for the rule REP/ACK 2/2. The green arrows will change to blue ones (pointing to the left).

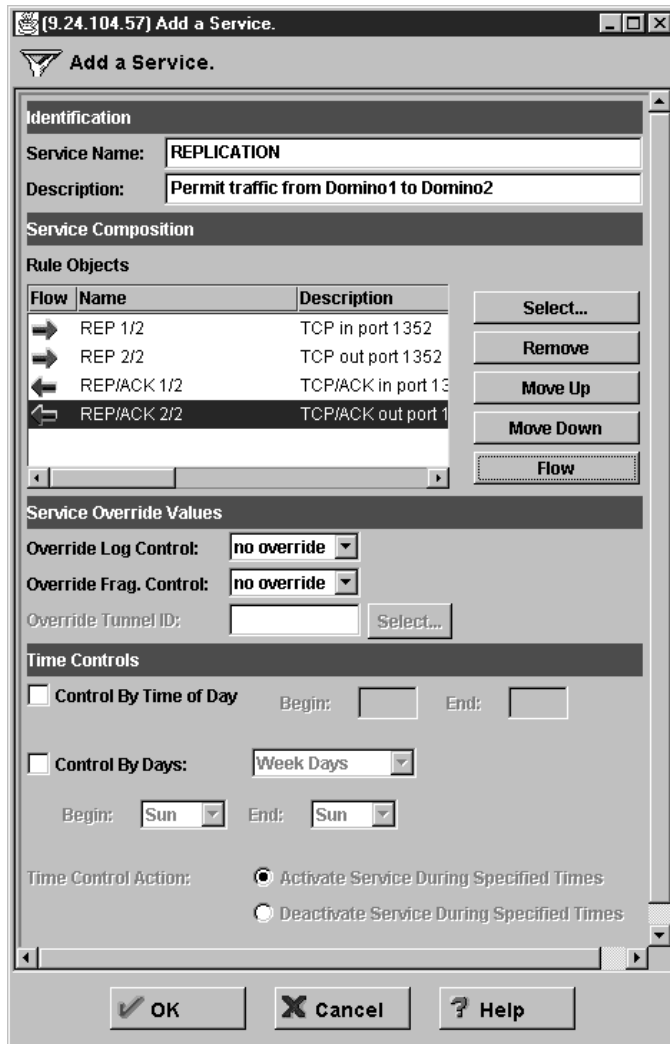


Figure 493. Changing Flow

7. Now click on **OK**.

Since we have the service configured, we can create the connection between the Domino1 server and the Domino2 server.

1. From the Configuration Client navigation tree select **Connection Setup** → **<NEW>** → **Open**.
2. Enter the name and a description of the connection.
3. For the source field click **Select** and choose the **Domino1** object.
4. For the destination field click **Select** and choose the **Domino2** object.
5. For the connections services click **Select**, choose the **REPLICATION** service from the list and click **OK**.

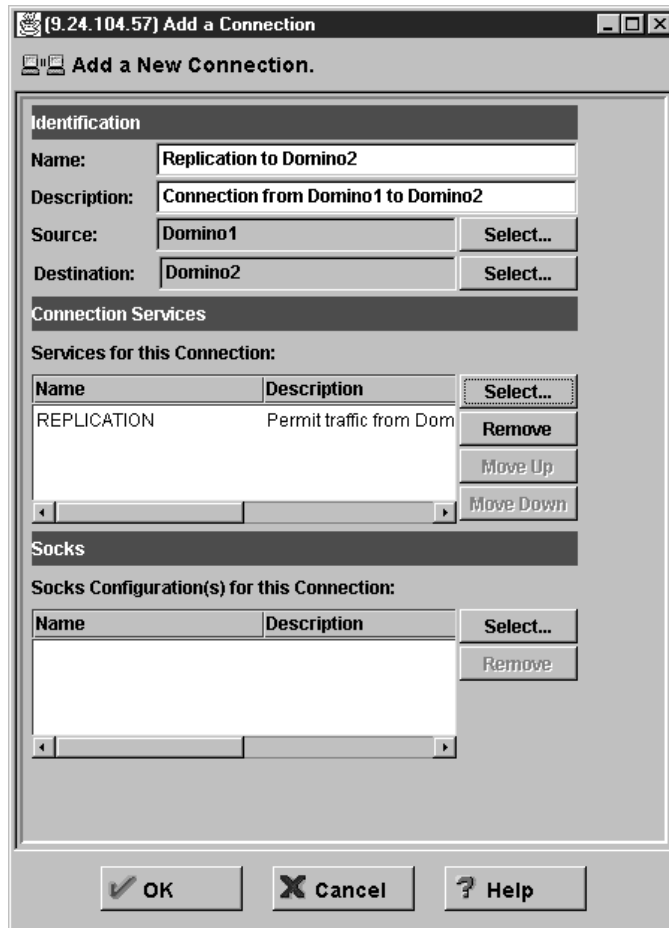


Figure 494. Domino1 to Domino2 Connection

6. Select **OK**.

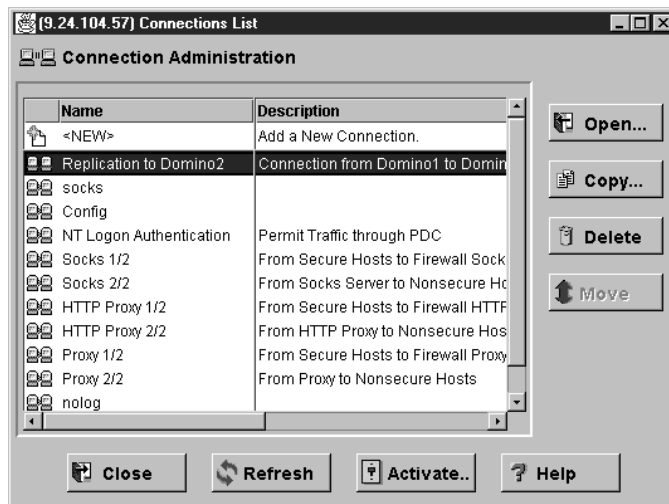


Figure 495. Connection List

The filter rules for the connections we have just created are shown in Figure 496 on page 473.

```
#Between Domino1 and Domino2
#      Service : REPLICATION
# Description : Permit traffic from Domino1 to Domino2
permit 9.24.104.77 255.255.255.255 192.1.1.2 255.255.255.255 tcp gt 1023 eq 1352 secure route inbound l=y f=y
permit 9.24.104.77 255.255.255.255 192.1.1.2 255.255.255.255 tcp gt 1023 eq 1352 specific(DMZ) route outbound l=y f=y
permit 192.1.1.2 255.255.255.255 9.24.104.77 255.255.255.255 tcp/ack eq 1352 gt 1023 specific(DMZ) route inbound l=y f=y
permit 192.1.1.2 255.255.255.255 9.24.104.77 255.255.255.255 tcp/ack eq 1352 gt 1023 secure route outbound l=y f=y
```

Figure 496. Filter Rules for the REPLICATION Connection

The first four rules permit the Domino1 server to communicate to the Domino2 server on port 1352 for replication of its databases. The other four rules permit hosts in the Internet to communicate to the Domino2 server on port 443.

After creating the connections you have to activate the filters rules.

Chapter 18. Testing The Configuration

After the IBM Firewall implementation is finished, the most important thing to do is to test the configuration, before somebody outside our organization does it for us. We have to convince ourselves and other people in the organization that there are no security holes in our configuration.

18.1 Introduction

To test the configuration we used the Internet Scanner SAFEsuite from Internet Security System (<http://www.iss.net>). Internet Scanner SAFEsuite is an analyzer tool that probes a network environment for vulnerabilities. It gives us information about each vulnerability found with location, in-depth description and the corrective actions we have to take to solve the problem. Internet Scanner SAFEsuite generates reports classified by vulnerabilities, services and hosts and we can choose if we want the report on text or HTML format.

Internet Scanner SAFEsuite scan services such as RPC, NFS, sendmail, anonymous FTP, NNTP, NetBIOS, and others. It also execute tests using Brute Force attacks and Denial of Service attacks (see Appendix A, "Security Issues Old and New" on page 597).

Internet Scanner SAFEsuite can be used to test networks connect to the Internet, stand-alone networks or machines. Is a good practice to run an analyzer tool periodically, weekly for example. You should run the analyzer tool every time you modify the firewall configuration.

18.2 Testing

We scanned on the non-secure interface connected to the Internet (see Figure 234 on page 258) with the firewall filters disabled, by enabling **Test IP Routing (debug only)** in the Security Policy panel. To scan the system with Internet Scanner do the following:

1. From the Internet Scanner main window, select **File** → **New Session**.
2. Enter a comment to the new session and click on **Next**.

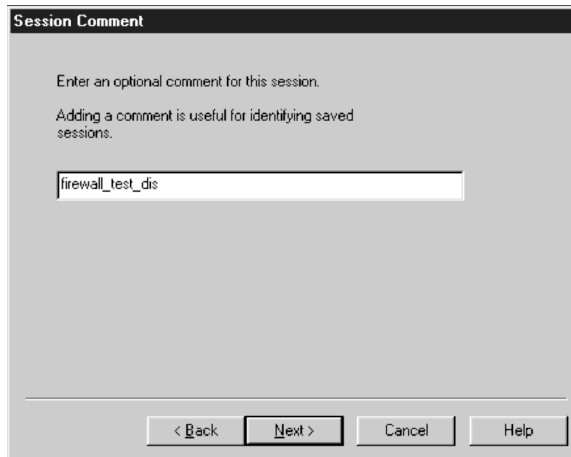


Figure 497. Session Comment

3. Choose the level of scanning you want by selecting a template and click on **Next**.

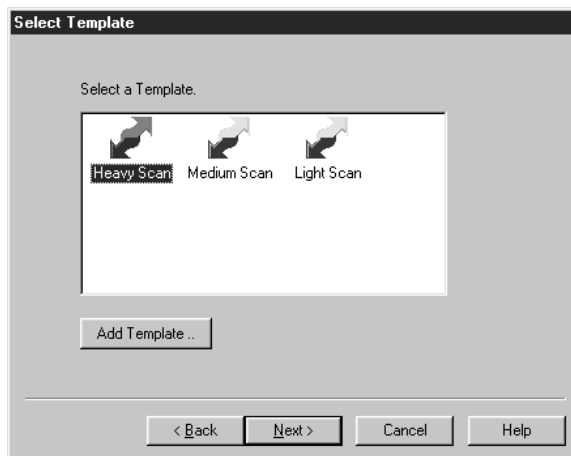


Figure 498. Selecting Template

4. Select a method to identify the host to be scanned and click on **Next**.
 - Use Host File:** Reads the IP addresses in the local
\\WINNT\System32\drivers\etc\ hosts file.
 - Use Command Line Facility:** Enter a range of IP addresses directly.
 - Have ISS Identify all valid hosts in your key range:** Select the hosts from
the license key.For our test we chose the **Use Command Line Facility**.

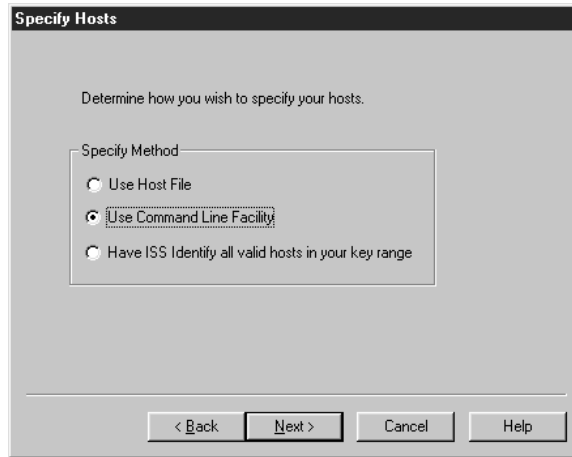


Figure 499. Specifying Hosts

5. Enter the address of the firewall machine and click on **Finish**.

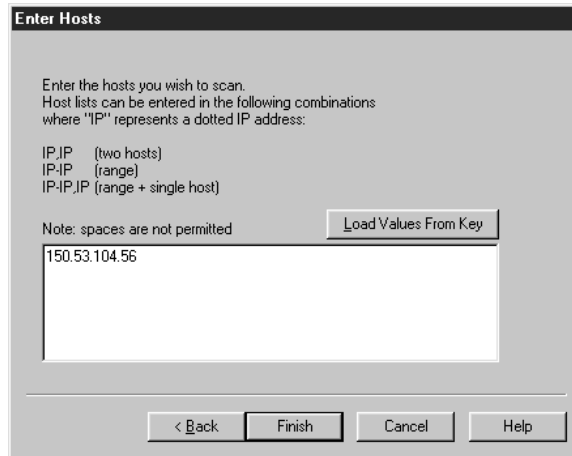


Figure 500. Enter Hosts

6. From the main window highlight the host address and select **Scan** → **Scan Now**.

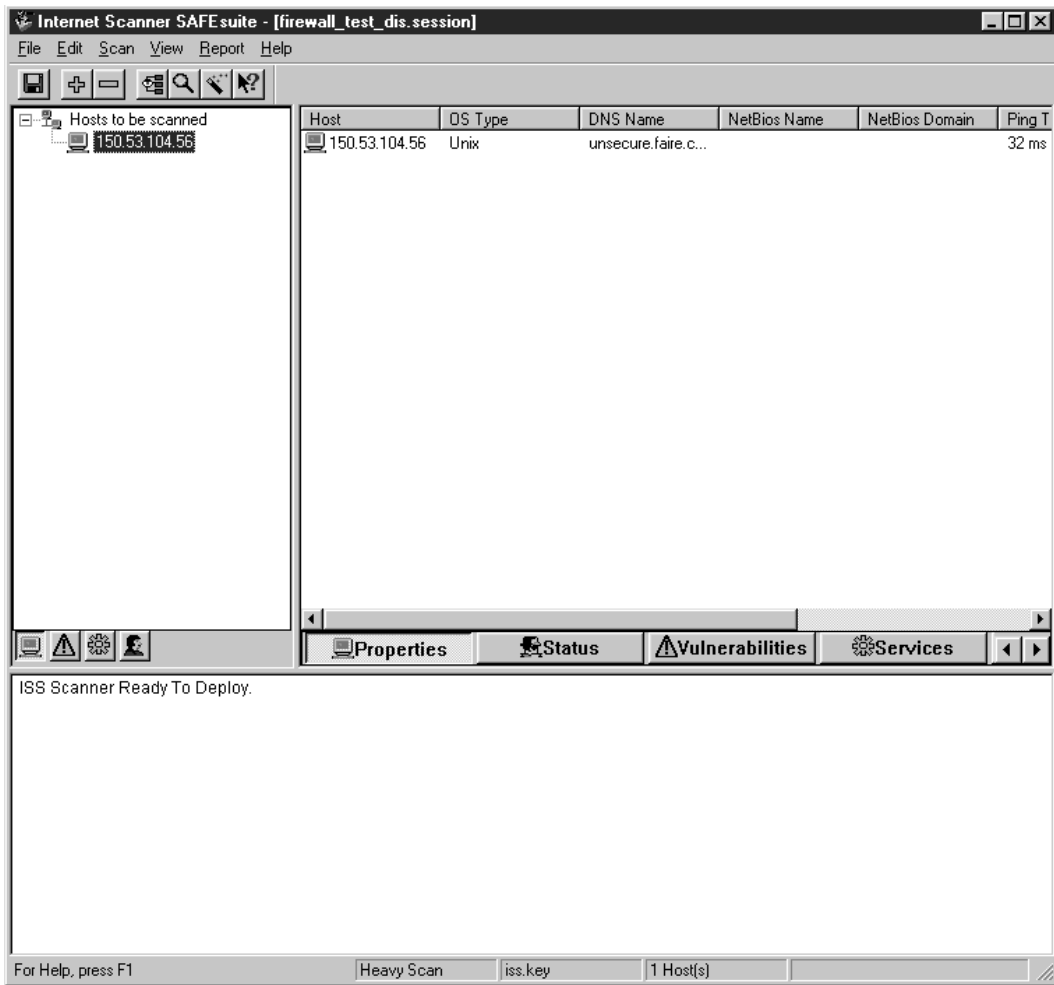


Figure 501. Selecting Host

The duration of the scanning process depends on the level of scanning we choose. For heavy scanning it could takes several minutes. After the scanning process the Internet Scanner main window will look like Figure 502 on page 479.

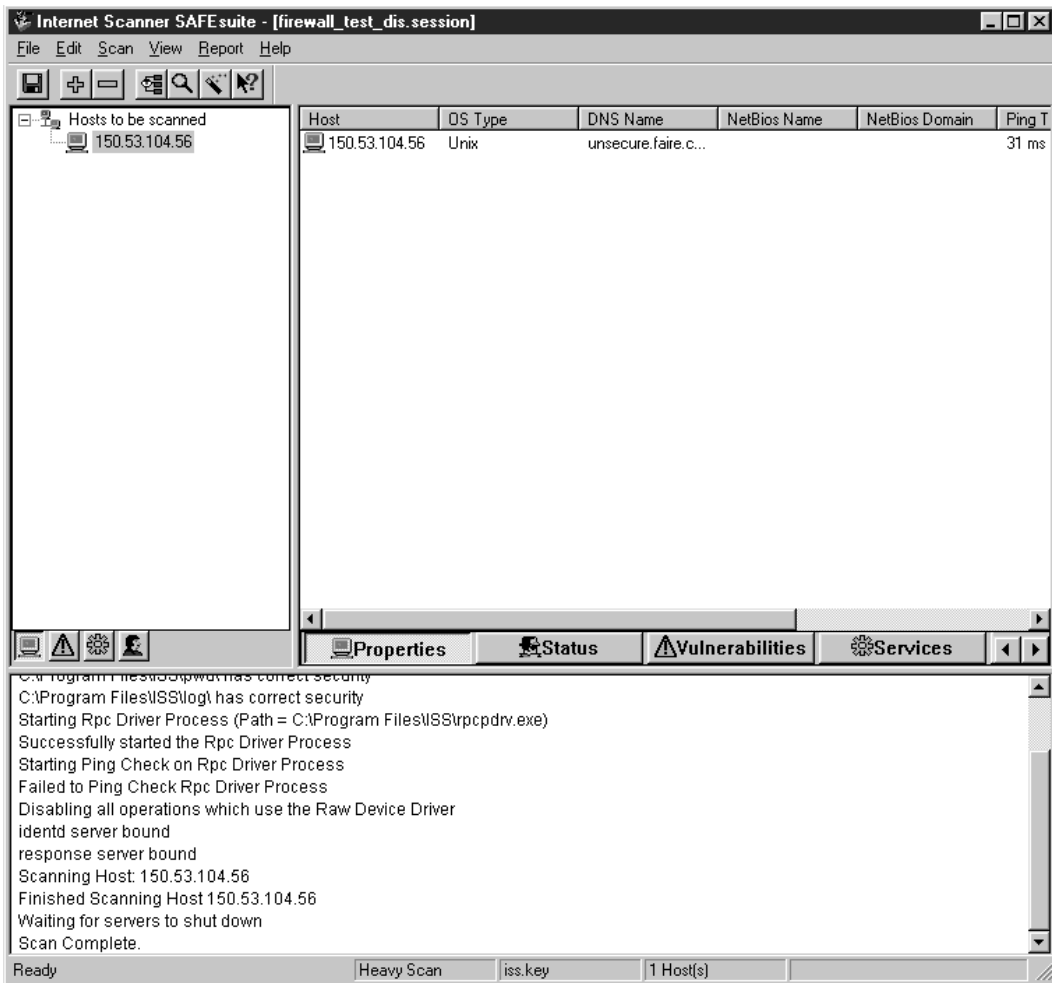


Figure 502. Scanning Complete

The OS Type field as informed by the ISS (Internet Security System) may not always be correct, as it is in our case.

Select the **Status** tab to see all the exploits made by the Internet Scanner and the initial time of each exploit.

We can see the scan status during or after the scanning process.

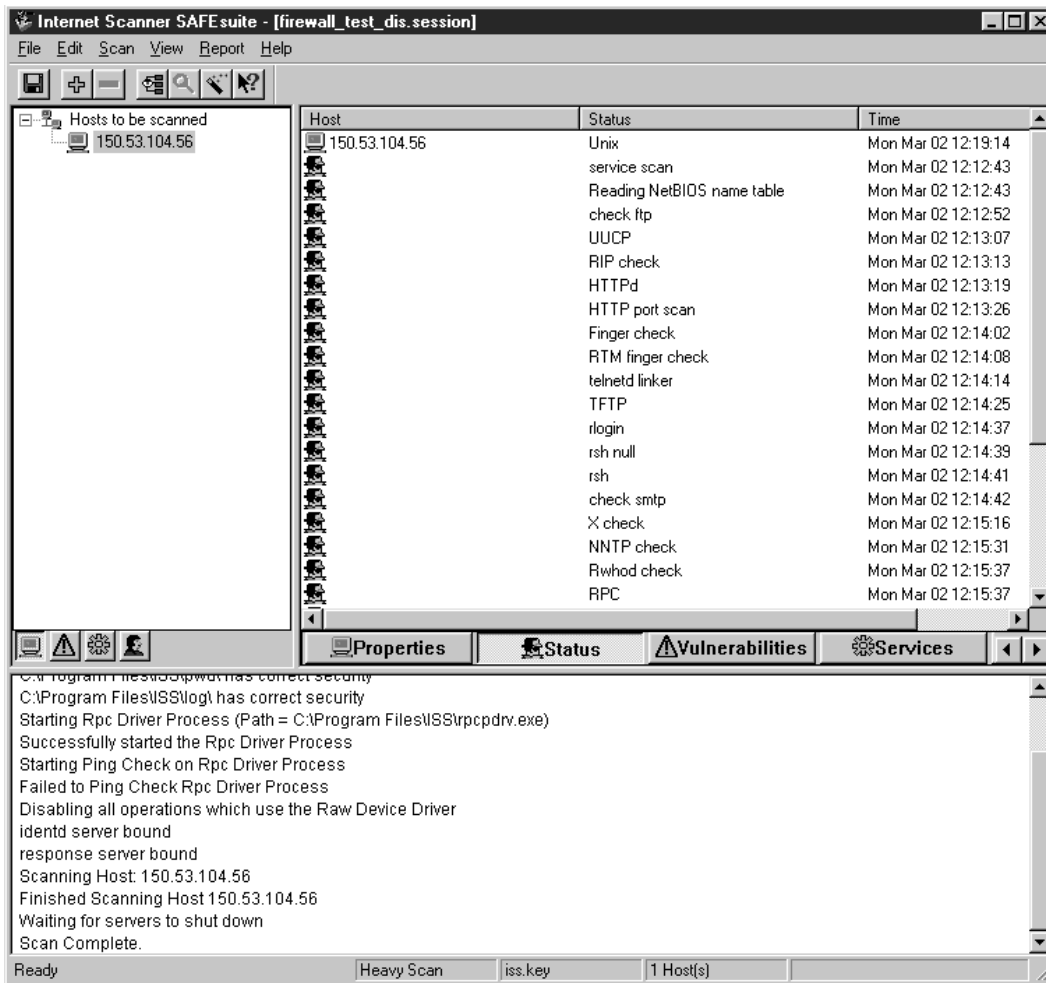


Figure 503. Scan Status

Select the **Vulnerabilities** tab to see the list of all the vulnerabilities found. We can use the mouse right button to see the description of the vulnerability including the action to be taken by the administrator.

As we can see in Figure 504 on page 481 the Internet Scanner found a traceroute vulnerability at our firewall configuration because we disabled all the firewall filters, letting any traffic through the firewall. Trace-routing a machine provides information concerning the topology of the network. It tells you whether you can reach a host and whether that host can answer, and also the route that packets take to reach the host. This information may allow an intruder to deduce trusted routers and other network information.

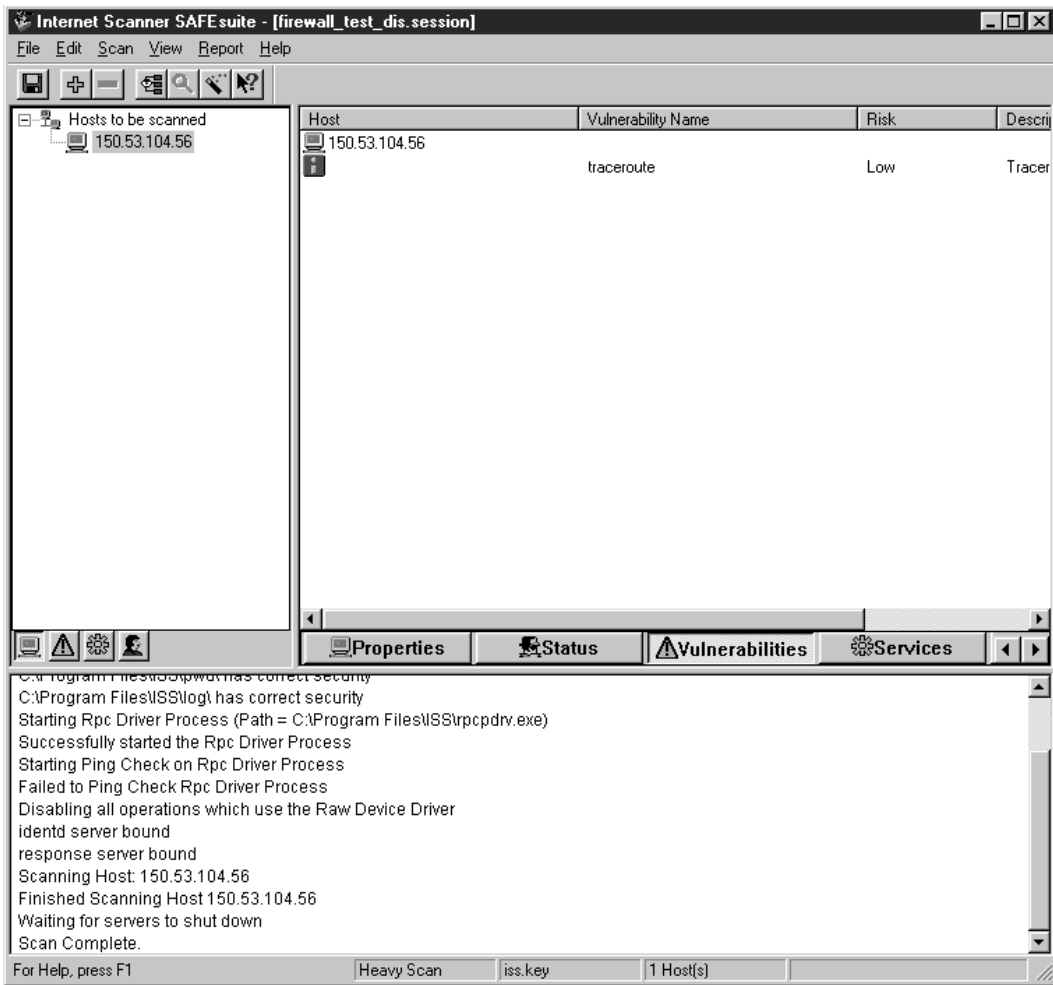


Figure 504. Vulnerabilities

Select the **Services** tab to see all the services that the Internet Scanner found in the firewall machine. These services were detected because the firewall was *open*.

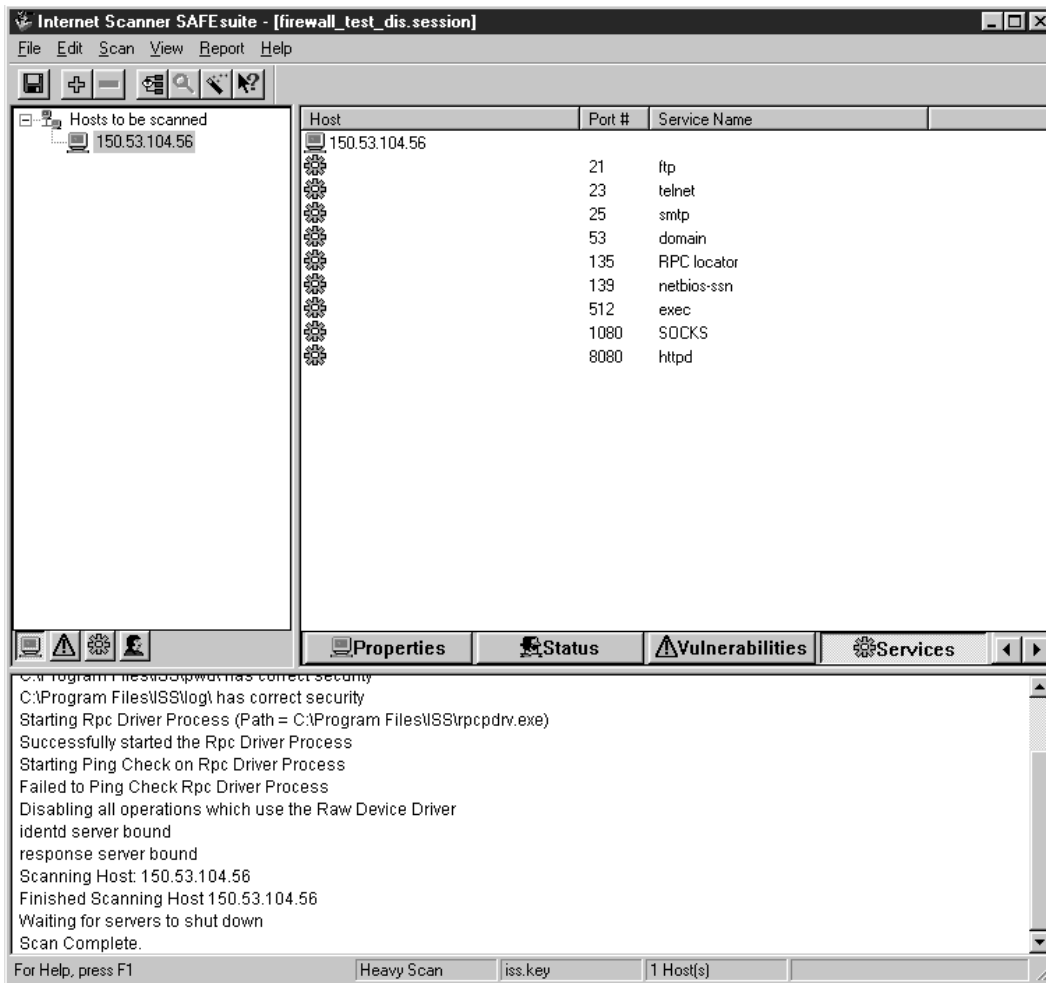


Figure 505. Services Found

Until now we have seen the scan results in the Internet Scanner main window. Now we will see how to generate reports to a text file.

1. From the Internet Scanner main window select **Report** → **Generate Report**.
2. Select which session you want to generate the report.
3. Select the **Options** tab to choose the text format.
4. Click on **Report**.

After the report generation, you will have a text file divided in reports as follows:

1. Summary Report of Vulnerabilities:
Provides a summary about the scan of our firewall.

Internet Security Scanner (C)1994-1997
4.3.2 Release
By Internet Security Systems, Inc.
Analysis Report

Summary Report of Vulnerabilities

Summary Information

Hosts Scanned: 1
Hosts Active: 1
Hosts Inactive: 0
Start Time: Mon Mar 02 12:12:14 1998
End Time : Mon Mar 02 12:19:15 1998
Total Time: 7 minutes 1 second
Session: firewall_test_dis.session
From Template: Heavy
Scan Completed Normally

Total Number of Vulnerability Risks: 1
Average Number of Vulnerability Risks per Active Hosts: 1.00

All Access NetBIOS share - Everyone Medium Risk: 0
All Access NetBIOS share - Guest High Risk: 0
All Access NetBIOS share found High Risk: 0
Autologon password readable High Risk: 0
Autologon enabled Low Risk: 0
DNS version vulnerable to denial of service Medium Risk: 0
IIS vulnerable to asp. bug Medium Risk: 0
Windows NT machine using LanManager authentication Low Risk: 0
Windows NT 3.5 detected Low Risk: 0
Windows NT 3.51 detected Low Risk: 0
Windows NT 4.0 beta detected Medium Risk: 0
NetBIOS share was found Low Risk: 0
Windows NT Network Monitor service running Medium Risk: 0
Windows NT Service Pack 3 not installed Low Risk: 0
Open NetBIOS share found Medium Risk: 0
Windows NT Remote Access service running Low Risk: 0
RPC Locator service vulnerable Medium Risk: 0
Open Ports via SOCKS Service High Risk: 0
Writable NetBIOS share - Everyone Medium Risk: 0
Writable NetBIOS share - Guest Medium Risk: 0
SNMP Community name world readable Low Risk: 0
Telnet no login required High Risk: 0
HKEY_CLASSES Writable - Everyone Medium Risk: 0
HKEY_LOCAL_MACHINE Writable - non-admin Medium Risk: 0
Chargen Service Medium Risk: 0
Trace Routing of Packets Low Risk: 1

Figure 506. Summary Report

The Internet Scanner gave us a summary of the vulnerabilities with the type of vulnerability, the level of risk (high, medium and low) and the number of hosts where the vulnerability was found. At the end of the report we can see the Trace Routing vulnerability found in one host.

2. Vulnerability Report:

Show all vulnerabilities found in our firewall configuration. Here again we can see the traceroute vulnerability.

Internet Security Scanner (C)1994-1997
4.3.2 Release
By Internet Security Systems, Inc.
Analysis Report

Report of Vulnerabilities

Summary Information

Hosts Scanned: 1
Hosts Active: 1
Hosts Inactive: 0
Start Time: Mon Mar 02 12:12:14 1998
End Time : Mon Mar 02 12:19:15 1998
Total Time: 7 minutes 1 second
Session: firewall_test_dis.session
From Template: Heavy
Scan Completed Normally

Information by Vulnerability

Trace Routing of Packets Low Risk

Traceroute allows you to determine the path a packet takes between two hosts. This information allows someone to determine network topology and routers on the network.

Was found on the following hosts:

IP Address: 150.53.104.56 Hostname: unsecure.faire.com
Route: 150.53.104.8 -> 150.53.104.56

Figure 507. Vulnerabilities Report

3. Services Report:

Provides information about all the services that are enabled in our firewall.

Internet Security Scanner (C)1994-1997
4.3.2 Release
By Internet Security Systems, Inc.
Analysis Report

Report of Services

Summary Information

Hosts Scanned: 1
Hosts Active: 1
Hosts Inactive: 0
Start Time: Mon Mar 02 12:12:14 1998
End Time : Mon Mar 02 12:19:15 1998
Total Time: 7 minutes 1 second
Session: firewall_test_dis.session
From Template: Heavy
Scan Completed Normally

Information by Service

Open Port

Was found on the following hosts:

IP Address: 150.53.104.56 Hostname: unsecure.faire.com
Port 8080 ("httpd" service) opened.
Port 1080 ("SOCKS" service) opened.
Port 512 ("exec" service) opened.
Port 139 ("netbios-ssn" service) opened.
Port 135 ("RPC locator" service) opened.
Port 53 ("domain" service) opened.
Port 25 ("smtp" service) opened.
Port 23 ("telnet" service) opened.
Port 21 ("ftp" service) opened.

Telnet Information

Telnet Banners allow intruder to quickly determine type of OS for system

Was found on the following hosts:

IP Address: 150.53.104.56 Hostname: unsecure.faire.com

SMTP Version

Was found on the following hosts:

IP Address: 150.53.104.56 Hostname: unsecure.faire.com

FTP Version

Was found on the following hosts:

IP Address: 150.53.104.56 Hostname: unsecure.faire.com
150.53.104.56: 220 ntfirewall FTP GATEWAY (Version 1.0) ready.

HTTP (WWW server) active

Was found on the following hosts:

IP Address: 150.53.104.56 Hostname: unsecure.faire.com
HTTPD Server: IBM-WTE/1.0

Figure 508. Services Report

4. Hosts Report:

Provides a summary about our firewall including services and vulnerability.

```
Internet Security Scanner (C)1994-1997
4.3.2 Release
By Internet Security Systems, Inc.
Analysis Report

Report by Hosts
Summary Information
  Hosts Scanned: 1
  Hosts Active: 1
  Hosts Inactive: 0
  Start Time: Mon Mar 02 12:12:14 1998
  End Time : Mon Mar 02 12:19:15 1998
  Total Time: 7 minutes 1 second
  Session: firewall_test_dis.session
  From Template: Heavy
  Scan Completed Normally

Information By Host

IP Address:      150.53.104.56
Hostname:       unsecure.faire.com

Host is Active
Host Types:
  UNIX
Services:
  Open Port
    Port 8080 ("httpd" service) opened.
    Port 1080 ("SOCKS" service) opened.
    Port 512 ("exec" service) opened.
    Port 139 ("netbios-ssn" service) opened.
    Port 135 ("RPC locator" service) opened.
    Port 53 ("domain" service) opened.
    Port 25 ("smtp" service) opened.
    Port 23 ("telnet" service) opened.
    Port 21 ("ftp" service) opened.
  Telnet Information
    Telnet Banners allow intruder to quickly determine type of OS
    for system

  SMTP Version

  FTP Version
    150.53.104.56: 220 ntfirewall FTP GATEWAY (Version 1.0) ready.
  HTTP (WWW server) active
    HTTPD Server: IBM-WTE/1.0
  Vulnerabilities:
    Trace Routing of Packets Low Risk
    Traceroute allows you to determine the path a packet takes
    between two hosts. This information allows someone to
    determine network topology and routers on the network.

    Route: 150.53.104.8 -> 150.53.104.56

Hosts Trusted by scanned Hosts

No trusted hosts found.
```

Figure 509. Hosts Report

Our next step was to scan the hosts in the internal network and the secure interface of the IBM Firewall, also from the external network.

To scan the internal network and the secure interface of the firewall, we followed the previous steps, but changed the IP address to be scanned to a range of IP

addresses from our license key given to us by Internet Security System (9.24.104.56 to 9.24.104.62). See Figure 510 on page 487.

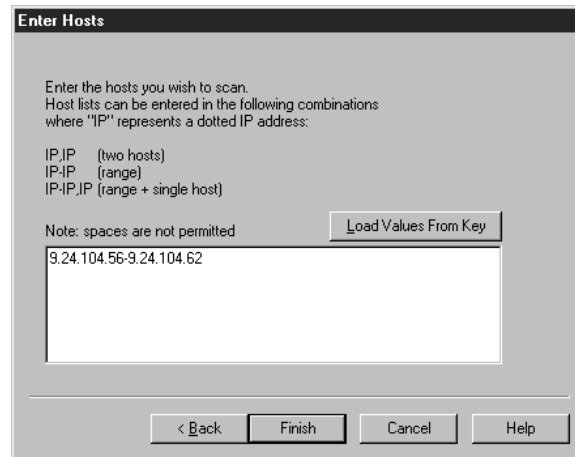


Figure 510. Enter Hosts

Following we show just a subset of the services and the hosts reports.

In the services report we could see all the services enabled in some internal hosts.

Internet Security Scanner (C)1994-1997
4.3.2 Release
By Internet Security Systems, Inc.
Analysis Report

Report of Services

Summary Information

Hosts Scanned: 7
Hosts Active: 7
Hosts Inactive: 0
Start Time: Thu Mar 05 16:32:52 1998
End Time : Thu Mar 05 16:41:31 1998
Total Time: 8 minutes 39 seconds
Session: Session8
From Template: Heavy
Scan Completed Normally

Information by Service

Open Port

Was found on the following hosts:

IP Address: 9.24.104.61 Hostname: wtrdc.itso.ral.ibm.com
Port 25 ("smtp" service) opened.
Port 23 ("telnet" service) opened.
Port 21 ("ftp" service) opened.
IP Address: 9.24.104.60 Hostname: rs60009.itso.ral.ibm.com
Port 6000 ("X" service) opened.
Port 1024 ("old_finger" service) opened.
Port 514 ("shell" service) opened.
Port 513 ("login" service) opened.
Port 512 ("exec" service) opened.
Port 135 ("RPC locator" service) opened.
Port 111 ("sunrpc" service) opened.
Port 37 ("time" service) opened.
Port 25 ("smtp" service) opened.
Port 23 ("telnet" service) opened.
Port 21 ("ftp" service) opened.
Port 19 ("chargen" service) opened.
Port 13 ("daytime" service) opened.
Port 9 ("discard" service) opened.
Port 7 ("echo" service) opened.
IP Address: 9.24.104.59 Hostname: WTR05090.itso.ral.ibm.com
Port 139 ("netbios-ssn" service) opened.
Port 135 ("RPC locator" service) opened.
Port 80 ("httpd" service) opened.
IP Address: 9.24.104.58 Hostname: rs600026.itso.ral.ibm.com
Port 1024 ("old_finger" service) opened.
Port 514 ("shell" service) opened.
Port 513 ("login" service) opened.
Port 512 ("exec" service) opened.
Port 135 ("RPC locator" service) opened.
Port 111 ("sunrpc" service) opened.
Port 37 ("time" service) opened.
Port 25 ("smtp" service) opened.
Port 23 ("telnet" service) opened.
Port 21 ("ftp" service) opened.
Port 19 ("chargen" service) opened.
Port 13 ("daytime" service) opened.

Figure 511. Services Report

In the hosts report we could see all the services and vulnerabilities found, sorted by host. Some vulnerabilities have a brief description.

```

Internet Security Scanner (C)1994-1997
4.3.2 Release
By Internet Security Systems, Inc.
Analysis Report

Report by Hosts
Summary Information
Hosts Scanned: 7
Hosts Active: 7
Hosts Inactive: 0
Start Time: Thu Mar 05 16:32:52 1998
End Time : Thu Mar 05 16:41:31 1998
Total Time: 8 minutes 39 seconds
Session: Session8
From Template: Heavy
Scan Completed Normally

Information By Host

IP Address: 9.24.104.61
Hostname: wtrdc.itso.ral.ibm.com

Host is Active
Host Types:
  UNIX
Services:
  Open Port
    Port 25 ("smtp" service) opened.
    Port 23 ("telnet" service) opened.
    Port 21 ("ftp" service) opened.
  Telnet Information
    Telnet Banners allow intruder to quickly determine type of OS
    for system

    OS/2 Version 2.3 (WTRDC)
    Enter your password:
  SMTP Version
    220 WTRDC.ITSO.RAL.IBM.COM Sendmail IBM OS/2 SENDMAIL VERSION 1.3.18/2.12

  FTP Version
    9.24.104.61: 220 WTRDC IBM TCP/IP for OS/2 - FTP Server ver 21:02:17 on Sep 25 1996 ready.
Vulnerabilities:
  Trace Routing of Packets ½Low Risk½
  Traceroute allows you to determine the path a packet takes
  between two hosts. This information allows someone to
  determine network topology and routers on the network.

  Route: 9.24.104.39 -> 9.24.104.61

IP Address: 9.24.104.60
Hostname: rs60009.itso.ral.ibm.com

Host is Active
Host Types:
  AIX
Services:
  Open Port
    Port 6000 ("X" service) opened.
    Port 1024 ("old_finger" service) opened.
    Port 514 ("shell" service) opened.

```

Figure 512. Hosts Report

Our last step was to enable the IBM Firewall filters and proceed with the tests also with the heavy scan level.

As shown in Figure 513 on page 490, Internet Scanner found no vulnerabilities in our environment.

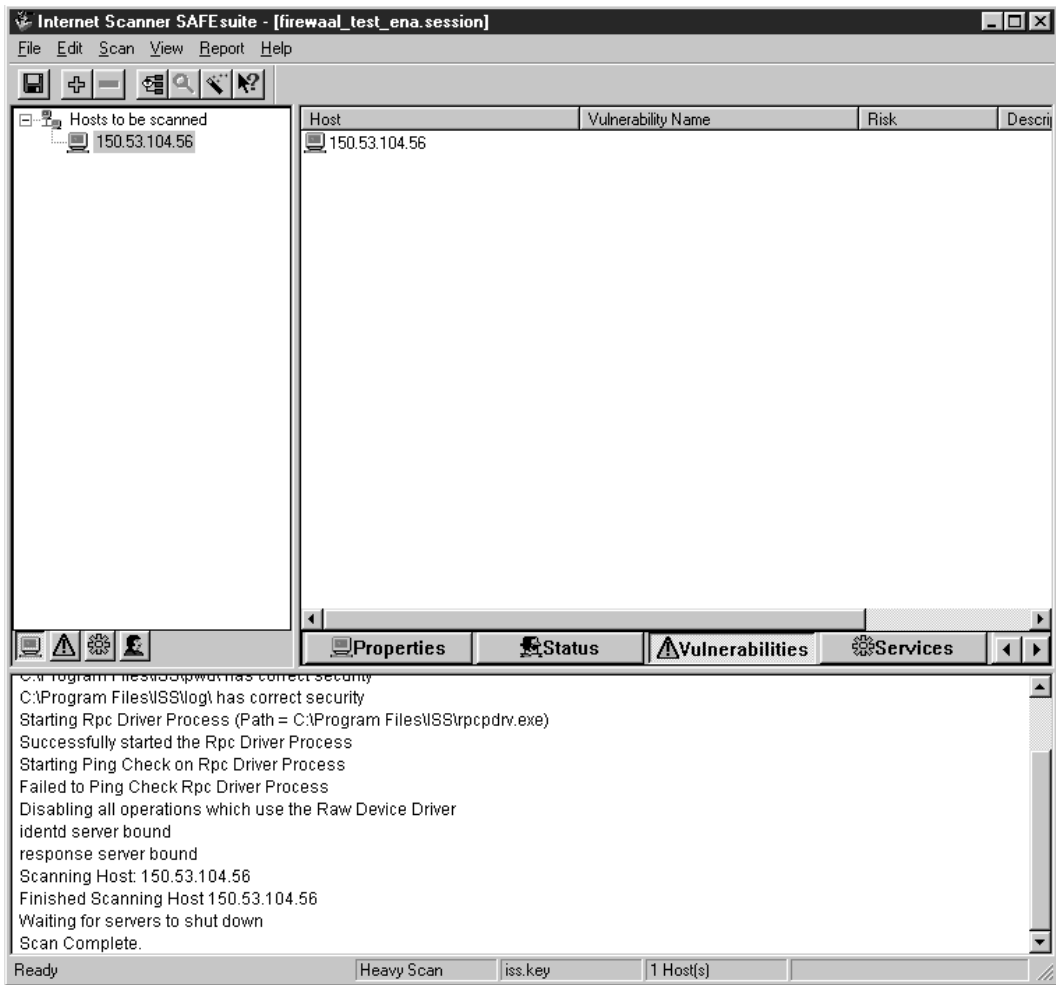


Figure 513. Vulnerabilities List

At the services list we could see the two services that were enabled: the SMTP service for e-mail coming from the Internet and the SOCKS service for SOCKSified client access from the Internet. In our case it was enabled to permit Host-on-Demand access from the Internet. See Chapter 9, “Examples of Rules for Specific Services” on page 161 for further information on specific services.

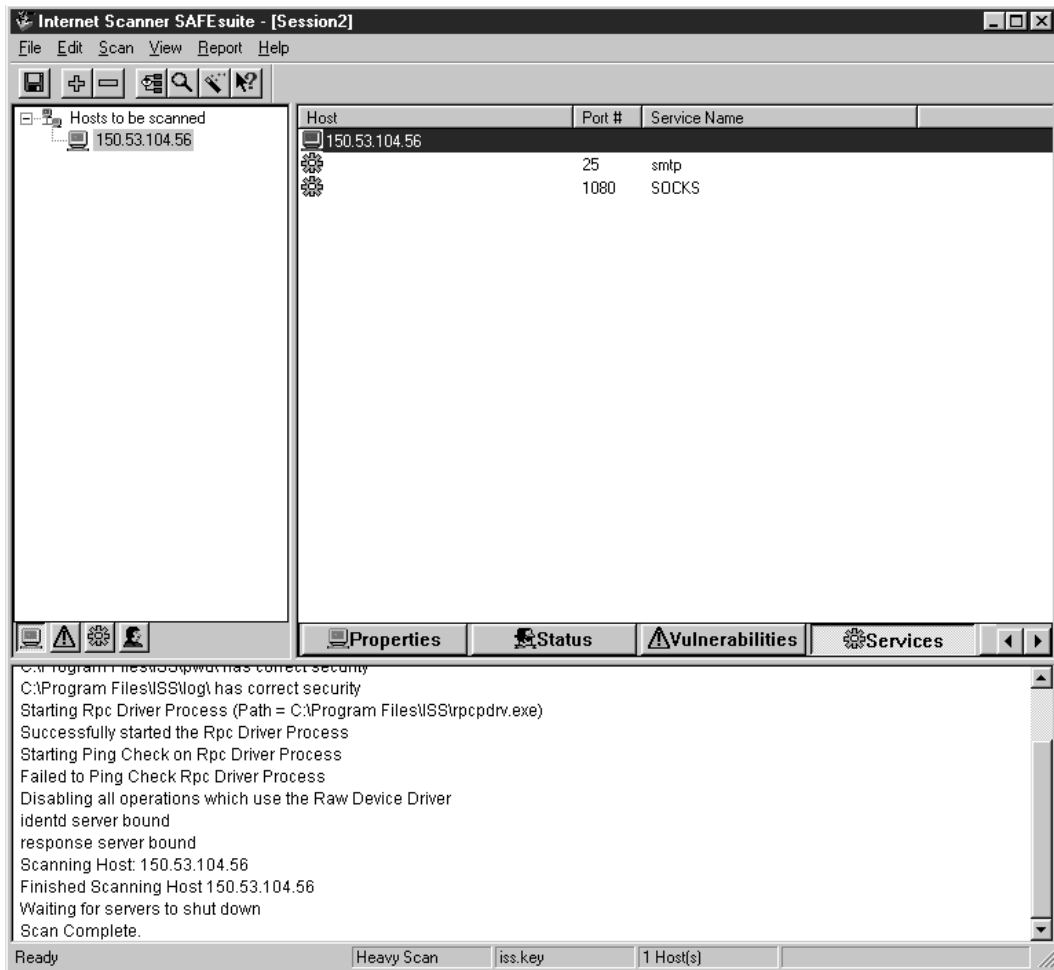


Figure 514. Services List

Following are the reports produced by Internet Scanner:

1. **Summary report** with the summary of the scan.

Internet Security Scanner (C)1994-1997
4.3.2 Release
By Internet Security Systems, Inc.
Analysis Report

Summary Report of Vulnerabilities

Summary Information

Hosts Scanned: 1
Hosts Active: 1
Hosts Inactive: 0
Start Time: Mon Mar 02 12:12:14 1998
End Time : Mon Mar 02 12:19:15 1998
Total Time: 7 minutes 1 second
Session: firewall_test_dis.session
From Template: Heavy
Scan Completed Normally

Total Number of Vulnerability Risks: 0
Average Number of Vulnerability Risks per Active Hosts: 0.00

All Access NetBIOS share - Everyone Medium Risk: 0
All Access NetBIOS share - Guest High Risk: 0
All Access NetBIOS share found High Risk: 0
Autologon password readable High Risk: 0
Autologon enabled Low Risk: 0
DNS version vulnerable to denial of service Medium Risk: 0
IIS vulnerable to asp. bug Medium Risk: 0
Windows NT machine using LanManager authentication Low Risk: 0
Windows NT 3.5 detected Low Risk: 0
Windows NT 3.51 detected Low Risk: 0
Windows NT 4.0 beta detected Medium Risk: 0
NetBIOS share was found Low Risk: 0
Windows NT Network Monitor service running Medium Risk: 0
Windows NT Service Pack 3 not installed Low Risk: 0
Open NetBIOS share found Medium Risk: 0
Windows NT Remote Access service running Low Risk: 0
RPC Locator service vulnerable Medium Risk: 0
Open Ports via SOCKS Service High Risk: 0
Writable NetBIOS share - Everyone Medium Risk: 0
Writable NetBIOS share - Guest Medium Risk: 0
SNMP Community name world readable Low Risk: 0
Telnet no login required High Risk: 0
HKEY_CLASSES Writable - Everyone Medium Risk: 0
HKEY_LOCAL_MACHINE Writable - non-admin Medium Risk: 0
Chargen Service Medium Risk: 0
Trace Routing of Packets Low Risk: 0

Figure 515. Summary Report

2. Vulnerabilities report: with no vulnerabilities found during the scan.

Once we enabled the filters of the IBM Firewall the Internet Scanner could not find the traceroute vulnerability as before.

```
Internet Security Scanner (C)1994-1997
4.3.2 Release
By Internet Security Systems, Inc.
Analysis Report

Report of Vulnerabilities
Summary Information
  Hosts Scanned: 1
  Hosts Active: 1
  Hosts Inactive: 0
  Start Time: Mon Mar 02 13:56:42 1998
  End Time : Mon Mar 02 14:08:24 1998
  Total Time: 11 minutes 42 seconds
  Session: firewall_test_ena.session
  From Template: Heavy
  Scan Completed Normally

Information by Vulnerability
```

Figure 516. Vulnerabilities Report

3. **Services report:** with the SMTP and SOCKS services found.

As we mentioned before, the SOCKS service was enabled to permit Host-on-Demand access from the Internet.

```
Internet Security Scanner (C)1994-1997
4.3.2 Release
By Internet Security Systems, Inc.
Analysis Report

Report of Services
Summary Information
  Hosts Scanned: 1
  Hosts Active: 1
  Hosts Inactive: 0
  Start Time: Mon Mar 02 13:56:42 1998
  End Time : Mon Mar 02 14:08:24 1998
  Total Time: 11 minutes 42 seconds
  Session: firewall_test_ena.session
  From Template: Heavy
  Scan Completed Normally

Information by Service

Open Port
  Was found on the following hosts:
  IP Address: 150.53.104.56  Hostname: unsecure.faire.com
  Port 1080 ("SOCKS" service) opened.
  Port 25 ("smtp" service) opened.

SMTP Version
  Was found on the following hosts:
  IP Address: 150.53.104.56  Hostname: unsecure.faire.com
```

Figure 517. Services Report

4. **Hosts report** with the information about our firewall where we can see the two services that were enabled from the outside world and no vulnerabilities found.

```
Internet Security Scanner (C)1994-1997
4.3.2 Release
By Internet Security Systems, Inc.
Analysis Report

Report by Hosts
Summary Information
  Hosts Scanned: 1
  Hosts Active: 1
  Hosts Inactive: 0
  Start Time: Mon Mar 02 13:56:42 1998
  End Time : Mon Mar 02 14:08:24 1998
  Total Time: 11 minutes 42 seconds
  Session: firewaal_test_ena.session
  From Template: Heavy
  Scan Completed Normally

Information By Host

IP Address:      150.53.104.56
Hostname:       unsecure.faire.com

  Host is Active
  Host Types:
    Unknown
  Services:
    Open Port
      Port 1080 ("SOCKS" service) opened.
      Port 25 ("smtp" service) opened.
    SMTP Version

  Vulnerabilities:

Hosts Trusted by scanned Hosts

No trusted hosts found.
```

Figure 518. Hosts reports

It's interesting to use other tools to test the firewall configuration. The Internet Scanner runs on the Windows NT platform, but if there is any UNIX machine that you can use, there are other tools that can be used.

In the following sections we give a brief description about the most used UNIX tools.

18.3 Network Security Auditor

Network Security Auditor, which is part of the IBM Firewall for AIX, is a tool that scans target machines and tries to exploit detected services of known weaknesses. For example, some obvious passwords are tried on services that ask for a password such as Telnet and FTP, or commands considered dangerous or risky on sendmail, such as DEBUG, VRFY and EXPN.

Network Security Auditor scans TCP, UDP and RPC services, for which you can select a predefined scan configuration or define your own scan configuration. Some examples of predefined scan configurations are:

- Default
 - tcp ports: 21, 23, 25, 111, 139, 512-514, 6000

- udp ports: 69,111,137,161
- rpc service: nfsmount
- Fulltcp
 - tcp ports: 1-65535
 - udp ports: 69, 111, 137, 161
 - rpc services: nfs,nfsmount,ypserv
 - options: tcp-seq-num, ftp-walk-tree
- Firewall
 - tcp ports: 1-65535
 - udp ports: 1-65535
 - options: ip-source-route, tcp-seq-num, ip-options, ftp-walk-tree
 - rpc services: all

Other scan types are: baseline, medium, standard and complete. See *Protect and Survive using IBM Firewall 3.1 for AIX, SG24-2577-02*, for more information about Network Security Auditor.

18.4 SATAN

SATAN (Security Administrator Tool for Analyzing Networks) is a tool designed by Dan Farmer and Wietse Venema to scan networks and pinpoint security problems. SATAN has been the subject of some criticism, because it is freely available (<ftp://ftp.win.tue.nl/pub/security/satan>). Some people feel that it is dangerous to make such a tool so widely available. The authors contend, however, that the dangerous crackers know all of SATAN's tricks already, so it is better to give the knowledge to network administrators, thereby letting them test their own defenses. Unlike other tools, it probes for security holes from outside and searches for a number of different security exposures, in addition to basic port scanning.

SATAN uses a Web browser for its graphical user interface and it generates the reports in HTML form so they can be viewed online within the Web browser. In order to run it you need PERL Version 5 or later.

Chapter 19. Backup, Restoration and Migration of the IBM Firewall

After you've meticulously set up and configured your firewall, you'll want to back it up. This way, if anything should happen to the firewall machine, you can recover quickly and with the minimum amount of effort. A backup is also helpful if you plan on making major (and untested) changes on the firewall.

Another reason would be to duplicate your firewall's filter rules on another firewall machine.

This chapter explains backup, restoration, and migration within the context of the configuration files of the IBM Firewall.

19.1 The IBM Firewall Configuration Files

Almost all of the IBM Firewall's configuration settings are neatly stored in the C:\Program Files\IBM\Firewall\config directory. The DNS files are the exception, as these are stored in the C:\WINNT\System32\DNS directory, along with one other parameter found in the registry. All the configuration files are simple text files and they do not take up much space. Thus, you can back these files up to a floppy.

In the sections that follow are a list of files you should, may, or should not back up.

19.1.1 The Configuration Map File

fwconfig.map contains a mapping of all the configuration files being used by the IBM Firewall. If you examine the map file, you will find some entries not present in the actual configuration directories. Do not be alarmed: some of these files are actually leftovers from development of the IBM eNetwork Firewall for Windows NT or are extras from the AIX version. The absence of these files has no effect on the operation of the IBM Firewall.

Your configuration files must be in the directories specified by fwconfig.map.

19.1.2 Connection Configuration Files

The connection configuration files define your network objects, rules, services, and connections.

fwobjects.cfg contains a list of all the network objects are defined on the firewall. This file contains hard-coded IP addresses.

fwrules.cfg contains a list of the pre-defined rules on the firewall, as well as rules you've defined.

fwservices.cfg contains a list of the services on the firewall. fwservices.cfg references the entries in fwrules.cfg.

fwsocks.cfg contains a list of SOCKS rules.

fwconns.cfg contains a list of connections used by the firewall. It references the entries in fwobjects.cfg, fwservices.cfg, and fwsocks.cfg.

fwfilters.cfg contains a list of the currently active filters. This file is generated from `fwconns.cfg` and the other previously mentioned files. This file contains hard-coded IP addresses. *Do not use this file in the restored firewall.* Instead, regenerate it from the IBM Firewall's GUI Configuration Client.

fwpolicy.cfg reflects the settings in the Security Policy dialog of the GUI. This file uses pre-defined services in `fwservices.cfg`.

fwadpt.cfg contains a list of the network adapters on the firewall as designated by their IP addresses and subnet masks. It also contains their secure and non-secure designations. While this file can be transferred to a restored firewall, we recommend that you do not. Instead, you should use the `fwadpt.cfg` file that the newly installed firewall generates. This file contains hard-coded IP addresses.

filters.active simply contains the date and time when the current set of filters was activated. There is no need to back up this file.

cfgfilt.output is similar to `filters.active`; contains the date and time when the current set of filters was activated. There is no need to back up this file.

19.1.3 Application Proxies and SOCKS Configuration Files

The following configuration files define the behavior of your application proxies and SOCKS service:

fwuser.cfg is a database of users.

fwtpproxy.cfg contains directives on the transparent proxy behavior of telnet and FTP.

fwhttp.cfg is the configuration file for the HTTP service.

fwTimeout.cfg contains idle timeout settings for telnet and FTP proxies.

socks5.conf is the configuration file for SOCKS generated by the GUI configuration client. You do not need to copy this file to the restored firewall; instead, you should regenerate it from the GUI.

socks5.header.cfg is the configuration file for SOCKS that you manually edit to take advantage of advanced settings. If you use only the GUI to configure SOCKS, there is no need to back up this file.

sockd.conf This file is for SOCKS v4, and is a holdover from the AIX version. There is no need to copy this file.

explode.cfg determines how the SOCKS service should act towards authentication. This strangely named file is manually edited.

Read Chapter 13, "Configuring Proxy Services and SOCKS" on page 293 for more information on these files.

19.1.4 Mail Settings File

`fwmail.conf` contains the settings for mail. You may simply copy it onto the restored firewall. This file contains hard-coded IP addresses.

19.1.5 DNS Settings Files

DNS files are located in C:\WINNT\System32\DNS. These files are:

- fwdns.lock
- fwnamed.boot.orig
- fwnamed.boot
- fwnamed.ca.save
- fwnamed.ca
- fwnamed.loc.save
- fwnamed.loc

Furthermore, the DNS server on the firewall boots from the registry, an additional item to take into consideration.

While you can back up these files and restore them on the new firewall, we believe it is better to regenerate DNS through the GUI. This is because when you set up DNS from the Configuration Client, it sets the resolvers and creates these files and the registry entry for you. All in all, using the Configuration Client in this case is less of a hassle than manual backup-and-restore of the DNS settings.

19.1.6 Virtual Private Network

There are three files created when configuring a VPN:

fwtunnel.cfg contains all the information about all the VPN tunnels

splidmgr.cfg contains the locally used Security Parameter Indexes (SPI) of our tunnel configurations

tunidmgr.cfg contains all tunnel id's

19.1.7 Network Address Translation Configuration File

There is only one configuration file for NAT called **fwnat.cfg** which contains the currently active NAT rules. If this file is missing there are no NAT rules defined.

19.1.8 Logging and Alert Configuration Files

There are three main configuration files for logging. These are:

syslog.conf defines the log facilities.

logmgmt.cfg defines which log facilities will be archived and how often.

fwtdfn.conf defines the alerts and corresponding actions on the firewall.

These files are explained fully in Chapter 16, "Logging, Monitoring, and Reporting" on page 409.

The log files in C:\Program Files\IBMFirewall\logs are automatically created once you restart the firewall, so there is no need to back them up for the purpose of restoration on another machine. (Though you might want to back them up as part of your archive procedures.)

However, there are no configuration files to run the **fwlogmgmt** command on a periodic basis (see 16.1.4.3, "Add Periodic Commands" on page 416.) **fwlogmgmt** is scheduled to run using the Windows NT at command. The settings for at are

found in the Windows NT registry. You could back up your registry simply for this reason, but if you only have a few entries for your scheduler, it would probably be easier (and safer) to reconstruct your at schedule files manually.

19.1.9 Modem and Pager Configuration Files

The modem and pager configuration files are explained in detail in 16.2.7, "Pager Setup" on page 431. These files are:

carriers.cfg contains a list of carriers. You only need to back this file up if you've defined entries for your pager company.

pager.cfg is the file that contains the pager information. You should back up this file, unless you use the methods explained in 16.2.6.3, "Configure Command Execution Monitors" on page 427.

modems.cfg contains a list of modems and their initialization strings. You only need to back up this file if you've defined custom modems.

Since there is no need to configure any Windows NT components for the modem and the pager, these files may be copied directly onto the restored firewall.

19.1.10 Configuration Server Files

The configuration server accepts connections from the configuration client, as described in Chapter 10, "Remote Administration Configuration" on page 221. There is only one settings file for the configuration server, **rscfile.cfg**. However, you should also copy any key files you may have generated for SSL connections. If you followed the instructions in 10.2.1, "Creating a Key File with mkkf" on page 226, these files should be found in the C:\Program Files\IBM\Firewall\config directory as well.

The files to save will thus be:

rscfile.cfg the configuration server configuration file

fwkey.kyr the key ring file

fwkey.sth the stash file

fwkey.cert the certificate file

19.2 Backing Up

The preceding sections should give you a good idea of which files are necessary to back up.

In principle, you should copy everything in the C:\Program Files\IBM\Firewall\config and C:\WINNT\System32\DNS directories. These files will supplement your records of the the firewall's configuration.

This backup should in turn be backed up by written documentation of your firewall's configuration.

19.3 Restoring

By a firewall restoration, we mean to say that a firewall machine will be configured to exactly the same parameters that the backed up firewall had. This includes the IP addresses and subnet masks of the interfaces of the firewall machine. The general procedure is as follows:

1. Reinstall Windows NT and the IBM Firewall, following the steps given in Chapter 4, "Installing the IBM eNetwork Firewall for Windows NT" on page 43.
2. Configure the network settings on the new firewall, using the the same network settings as the old firewall.
3. Copy all the files from the old C:\Program Files\IBM\Firewall\config directory into the new config directory. The path for the configuration directories must be exactly the same as these are all mapped out in the fwconfig.map file.
4. Reboot the firewall so that the new settings will take effect.
5. Open the GUI Configuration Client and log on as firewall administrator.
6. Review the firewall settings against your written documentation. Enter new DNS information as well.
7. Create new archive management schedules using the Windows NT Schedule service.

19.4 Migration

What about the case when you want to use the configuration of one firewall on another firewall? We call this type of activity *migration*.

There are several restrictions to this, chiefly on the differences between the IP addresses these two machines will be using in their configurations. And to migrate between two radically different configurations, for example, from a two-adapter firewall to a three-adapter firewall, would be far too complex an undertaking. Far better to rebuild everything from the ground up.

However, if the two configurations are quite similar, it might be a worthwhile undertaking. From our point of view, the only practicable reason for doing so would be to take advantage of well-developed filter connection rules, proxy definitions, and log management settings already in place on a firewall.

Such an activity should only be carried out after you have studied and understood the implications of the configuration files you will be working with. The descriptions given in 19.1, "The IBM Firewall Configuration Files" on page 497 can be used as a guide to determine which files you can and should use. In particular, be careful about files with hard-coded IP addresses.

Chapter 20. SecureWay Network Dispatcher

Handling a huge load of traffic can be a very tough problem. IBM managed the high load of the Nagano Olympics Web site by using its SecureWay Network Dispatcher, a multipurpose load balancing facility. This software was able to handle a peak of up to 104,000 hits per minute.

Of course not only Web servers can be hit with high loads - an enterprise firewall is often groaning under the traffic it has to handle. And because there is a limit to adding more RAM or a better or more CPUs, it would be much easier to scale a firewall by adding another machine, much like what is done with Web servers.

So this chapter shows you some very interesting examples of what you can do with the ND and also the common problems you may run into under Windows NT.

The IBM SecureWay Network Dispatcher Version 2.1 is part of the IBM WebSphere Performance Pack Version 2.0.

We recommend reading *IBM WebSphere Performance Pack Usage and Administration*, SG24-5233 to learn more about the ND. The redbook on IBM WebSphere Performance Pack Version V2.0 will soon be published.

20.1 The Basics of ND

To introduce you to ND and its capabilities with firewalls we assume an environment as in Figure 519.

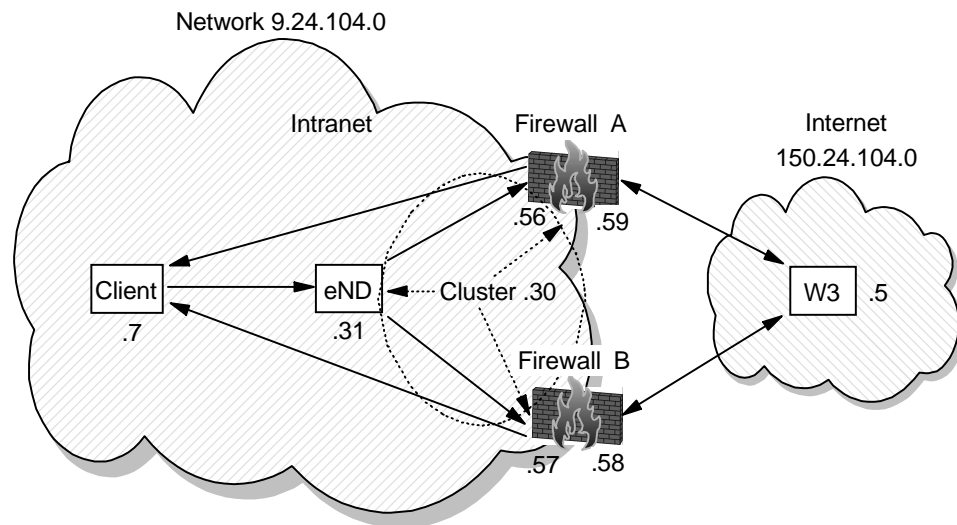


Figure 519. A Simple ND Environment in Conjunction with Firewalls

For a better understanding we will show you a possible flow from the client in the Intranet to the W3 server in the Internet. There will be no technical description yet, we will just go with the flow. Try to follow the arrows with the connected lines to understand what happens in each step.

The following actions take place when the client wants to access the W3 server with its Web browser. The browser is configured to use the ND as an HTTP proxy:

1. The client Web browser sends an HTTP proxy request to the ND.
2. The ND receives the request and decides whether to use firewall A or B. Let's assume it will choose firewall A and redirects the request to firewall A, not by sending a HTTP proxy request by itself, but by resending the client's request to do so.
3. Now firewall A receives a request for HTTP proxy, but it looks like it is coming directly from the client. The firewalls A HTTP proxy will now process the request and sends a request to the W3 server.
4. The W3 server answers the request it just received from the firewall A; sends its answer back to Firewall A.
5. On firewall A the HTTP proxy receives the W3 server's answer and will send it back to the requester. But because it seemed to firewall A like the client was requesting something, it will send the answer directly back to the client and not to the ND.
6. At the last step the client receives the answer to its request and it looks to it like it gets the answer from the ND but you know it doesn't though the client is not able to see a difference.

Right now you know that the ND will redirect requests to another machine that will take care of this. How can this goal be reached?

When you have a closer look at Figure 519 on page 503 you will recognize an ellipse that includes the ND and both firewalls. This symbolizes the so-called cluster - a group of some machines that have something in common. In our case they have the same IP address. Of course not all of them will answer to this IP address when you send a ping to it, but just only the ND. How this trick is achieved will be shown later.

Now we have two firewalls but only one ND, so this machine becomes the single point of failure. Section 20.4, "High Availability" on page 515 discusses how to avoid this problem.

20.2 Technical Background

Now we are going into a more detailed description of how we built such a cluster.

20.2.1 Installation and Configuration of the ND

The machine where we had the ND installed is a Windows NT Server; in our case, Windows NT4.0 SP3. There was nothing special installed on this server, but we had to make sure that IP forwarding was turned off. The ND will work with this feature turned on, but you will get very strange effects like enormous delays on IP traffic.

For the ND installation we chose to install just the SecureWay Network Dispatcher (no File Sharing, Caching and Filtering or the Common Configuration), but before we could proceed we had to install Sun JDK Version 1.1.6, which is included with the WebSphere Performance Pack. After installing the JDK we had to reboot the machine and start the setup again. When we had the choice to select the available components we took only the Load Balancing facility; Interactive Session Support or Content Base Routing are not necessary for our environment. There was a

message at the end of the installation that we should reboot the machine and so we did.

Configuring the ND with the supplied GUI is easy. Most of the actions you can do here is by right-clicking an object and selecting an action that appears.

After starting the GUI we connected to our host (that is, the machine where the ND is installed) by right-clicking on the Dispatcher line and then we started the so-called executor. The next step was to add a cluster - we took an IP address that was not yet allocated by another host: 9.24.104.30.

We added a port: 8080 for the HTTP proxy. To this new port we added the IP addresses of firewall A's and B's secure network adapter, that is 9.24.104.56 and 9.24.104.57.

In addition to this we added several other ports and each time we added both firewalls to this too. You can see the result in Figure 520.

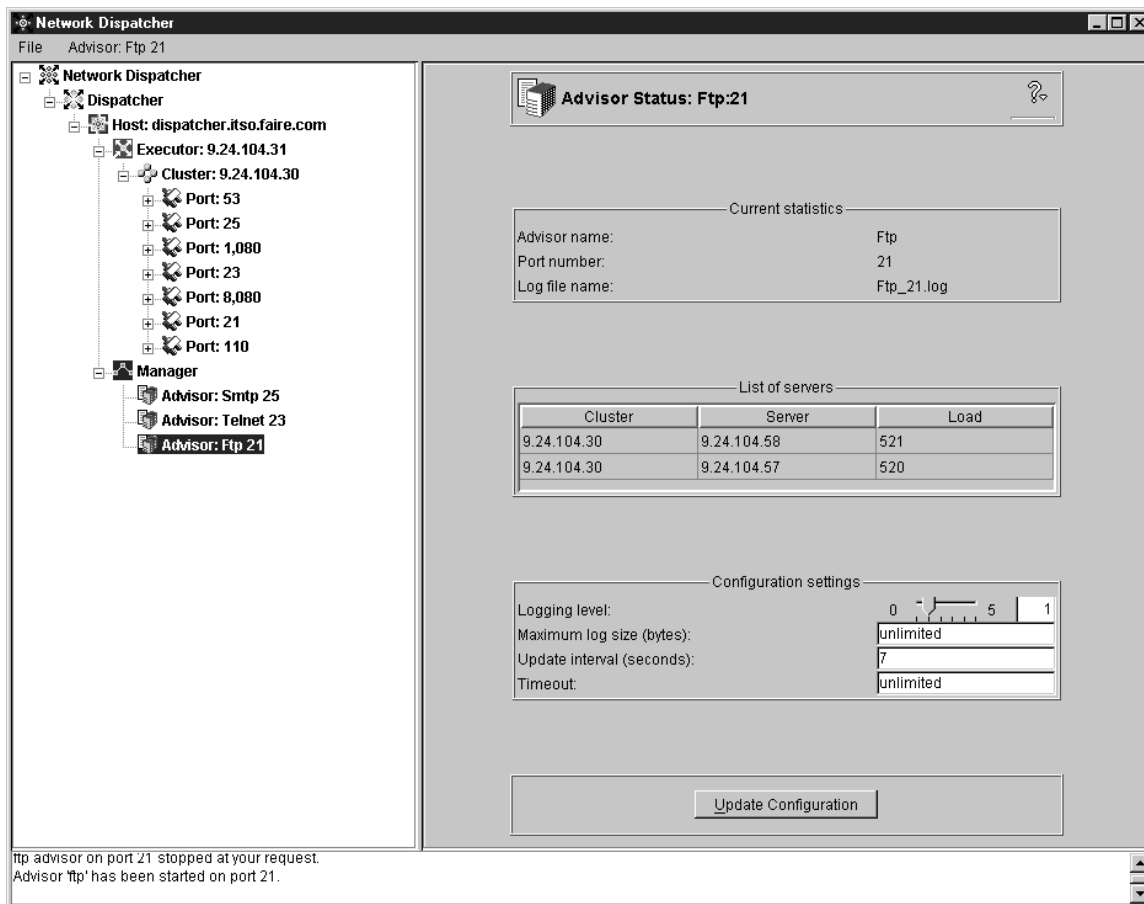


Figure 520. Example Configuration of the Network Dispatcher

The load balancing component of the ND has some very interesting mechanisms to automatically provide you with the best server available. One of these mechanisms is to give servers a weight, and the higher the weight, the more it counts. This means a server with a greater weight will be chosen before that with a lower weight.

When you add a new server to a port, ND will try to reach the server and if it is not possible, it will set its weight to 0. This means that this machine will not be used at all. To activate this feature we had to start the manager. When you right-click on the host line (the line with the machine name) you have the ability to start it. When the manager is running it will send a ping to the servers defined in a cluster. Without the manager you will have only a basic round robin load balancing without checking if a server is available. This means that we defined a rule on each firewall to allow ping requests from the ND to the firewall's secure interface. Now, that a machine is reachable doesn't mean that the service it should handle is working. That is the job of the advisors.

The manager that does the basic reachable ping also controls these advisors. We added a few advisors to the manager for a better control of the services on each port (Port 21 FTP, 23 Telnet, 25 SMTP, 53 DNS, 110 POP3, 1080 SOCKS and 8080 HTTP Proxy as shown in Figure 520 on page 505).

The manager will allow a better load balancing because it is able to set the proportions as seen in Figure 521.

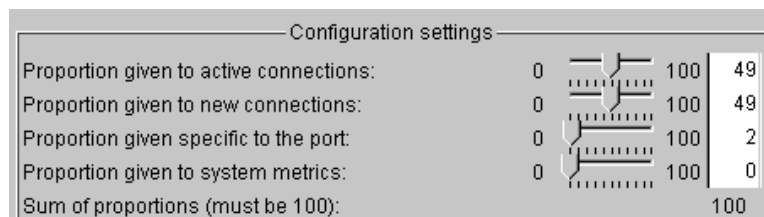


Figure 521. Proportion Settings of the Manager

Basically an advisor checks the current state and speed of a service on each server and so it is able to influence the weight of those servers. An advisor can be a simple ping, but in the case of HTTP it sends a specific request to this service. For the Web Traffic Express Http Proxy (WTE) there is a special implementation on the proxy side for the WTE advisor.

We used the proportion settings as recommended in the users guide, and they worked fine.

By doing so we run across a problem with the initial release of the IBM eNetwork Firewall for Windows NT; we added an advisor for HTTP that pointed to port 8080. Right after we activated this, the HTTP proxy on both firewalls crashed; we saw a Dr.Watson saying so on the screen. The problem has been located and it should be solved in the refresh 3.3.1 of the IBM eNetwork Firewall for Windows NT.

As the advisor is just an additional help, to get a better control of the services it is not a must to have. This means that our HTTP proxy scenario will do fine. Actually, there are a lot of services where no advisor is defined, but as said before it's an additional help, not a requirement.

The last step in order to get the ND active is to tell it to what IP address it has to listen. Just adding the cluster IP address in the GUI will not do the job. We opened a command line and typed:

```
ndconfig tr0 alias 9.24.104.30 netmask 255.255.255.0
```

We used tr0 because of the token-ring environment in our tests. Also we do not define the IP address in the network panel; the ND is capable of listening to the network by itself.

Now we saved the configuration as "firewall".

There is still one more step in the ND setup process. We need to start a batch to invoke some commands that tell the ND to load the configuration and set the cluster IP address alias. This is quite difficult because Windows NT does not support such a thing like the autoexec.bat by default. We found a utility named AUTOEXNT on the Internet that is part of the Windows NT Resource Kit. We checked the Resource Kit and found the program, but it seemed to be a little old. The archive contained a text file on how the program is to be installed. It said that there is a program called INSTEXNT, but the program name is now INSTSRV. This was the only difficulty we had with installing the service.

When we were done there was a AUTOEXNT.BAT in the system root system32 directory. We changed the file to the following:

```
@ECHO OFF
SET JDKPATH=C:\jdk1.1.6\bin
SET IBMNDPATH=C:\WSP\IBM\nd
SET ND_AFFINITY_PORT=10005
SET ND_RMIPORT=10099
SET END_ACCESS=-DND_CLIENT_KEYS_DIRECTORY=%IBMNDPATH%\admin\keys\dispatcher
SET CLASSPATH=%IBMNDPATH%\dispatcher\lib;%IBMNDPATH%\dispatcher\lib\ibmnd.jar
SET CLASSPATH=%CLASSPATH%;%IBMNDPATH%\admin\lib\ChartRuntime.jar
SET Path=%PATH%;%IBMNDPATH%\dispatcher\bin;%IBMNDPATH%\dispatcher\lib;%JDKPATH%
CALL ndcontrol.cmd file load firewall
NDCONFIG tr0 alias 9.24.104.30 netmask 255.255.255.0
EXIT
```

All the SET commands will ensure that the environment is correct. The CALL command will instruct the ND to load our previously saved configuration and also start the executor and the manager. In last line, the NDCONFIG will set the cluster IP address alias.

With this setup, ND will continue to work after a reboot or a possible power failure.

20.2.2 Configuration of the Firewall

To have your firewalls ready to take part in the ND cluster is not that much work at all. You will have to install an additional network adapter which is called the "MS Loopback Adapter". This is not a physical adapter, just a software to emulate one. We proceeded as if we wanted to install a normal network adapter by adding a new one in the network configuration panel. In the list of possible adapters we selected the MS Loopback Adapter; As for frame type, the default 802.3 is fine. Next we entered the cluster IP address 9.24.104.30 to the loopback adapter (if you experience problems with the SOCKS server see 20.2.4, "Load Balancing with Different Services" on page 508).

After rebooting, we detected an additional route to our secure network 9.24.104.0 (first line in the route list below). This caused problems because the loopback adapter is just a local device and not physically connected to our secure LAN.

Network Address	Netmask	Gateway Address	Interface	Metric
9.24.104.0	255.255.255.0	9.24.104.30	9.24.104.30	1
9.24.104.0	255.255.255.0	9.24.104.56	9.24.104.56	1
9.24.104.56	255.255.255.255	127.0.0.1	127.0.0.1	1
9.255.255.255	255.255.255.255	9.24.104.56	9.24.104.56	1
127.0.0.0	255.0.0.0	127.0.0.1	127.0.0.1	1
150.24.104.0	255.255.255.0	150.24.104.59	150.24.104.59	1
150.24.104.59	255.255.255.255	127.0.0.1	127.0.0.1	1
150.24.255.255	255.255.255.255	150.24.104.59	150.24.104.59	1
224.0.0.0	224.0.0.0	150.24.104.30	150.24.104.30	1
224.0.0.0	224.0.0.0	150.24.104.59	150.24.104.59	1
224.0.0.0	224.0.0.0	9.24.104.56	9.24.104.56	1
255.255.255.255	255.255.255.255	9.24.104.56	9.24.104.56	1

As these routes are created after every reboot we removed this unnecessary route with the AUTOEXNT as we did before on the ND machine:

```
@ECHO OFF
ROUTE DELETE 9.24.104.0 MASK 255.255.255.0 9.24.104.30
EXIT
```

This had to be done on both firewall machines.

Checking the firewall configuration we found out that it detected the new adapter and its status was non-secure. After changing it to secure we created a new network interface object for it and added this interface to the secure interface group. The rest of the connection and filter rules needed no change. Our last step was to check if the firewalls were configured to behave equally on all services.

20.2.3 How it works

We will now have a more detailed description on how the cluster will work. You have recognized the difference between defining the cluster IP address alias on the ND and on the firewalls. The reason is simple - when there is a IP request pending on the network for the cluster IP address, only one machine should be responsible for this address. Only ND will respond to an ARP request for the MAC address of the cluster IP address. We reach this goal with the ND software itself which will listen to the network and catch packets for the cluster IP address defined with the NDCONFIG command. The Network Dispatcher will make a decision where this packet has to be sent, and will just exchange the MAC address of the ND machine with that of the firewall that should receive this packet; it then sends it back to the network. The destination firewall will pick up the packet because it is for its MAC address and since it also has the cluster IP address defined via the loopback adapter it is able to handle it.

At first the packet finds its way by the IP address and then the MAC address is rewritten to reach its final destination.

```
<Client MAC addr.:ND MAC addr.> -> <Client MAC addr.:Firewall MAC a ddr.>
<00-06-29-A9-72-0B:40-00-52-00-52-04> -> <00-06-29-A9-72-0B:08-00-5A-B9-6 9-05>
```

20.2.4 Load Balancing with Different Services

We described the basic flow in section 20.1, "The Basics of ND" on page 503. We are ready to use our client's browser to see what happens when we try to use our cluster as a HTTP proxy.

Figure 522 on page 509 shows the monitor for port 8080 which can be started when you right-click on the port number.

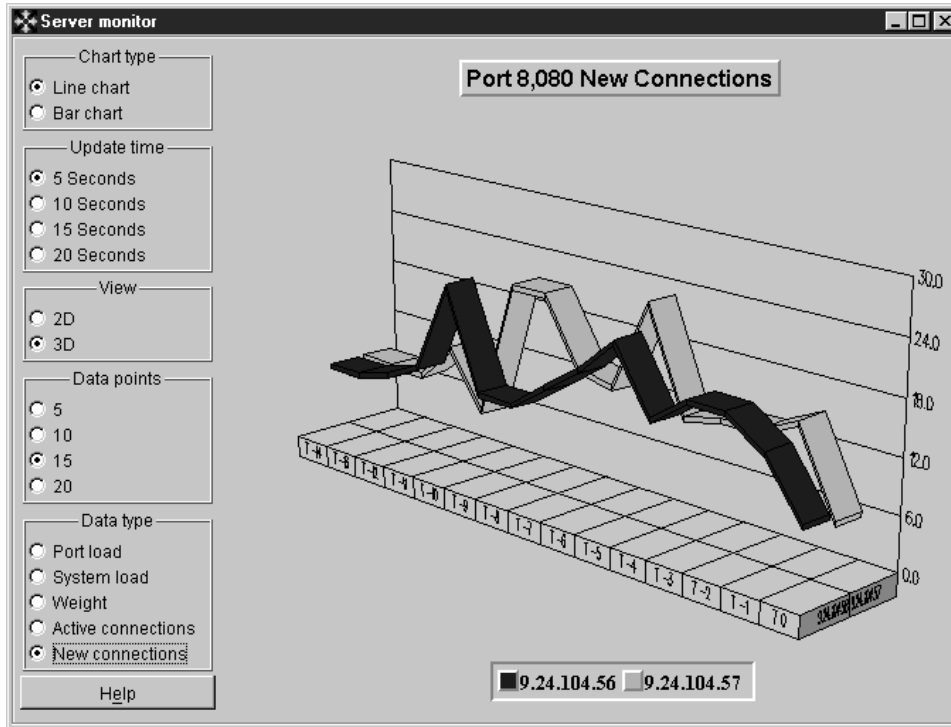


Figure 522. Monitoring New Connections on Port 8080 (HTTP Proxy)

We selected to view new connections and then we used the client's Web browser to access the Web server on the non-secure side. The client's performance was pretty good, and as Figure 522 shows the ND is load balancing the connections very well. Checking both firewall logs we were able to confirm that both firewalls had been used.

As this test was successful we tried some other ports like FTP, Telnet, POP3, SMTP and so forth. It looked like the ND and firewall combination was handling everything very well but we experienced some strange behavior with DNS and SOCKS.

Because DNS is a TCP/UDP-based protocol and UDP is a connectionless protocol, each packet contains a bit (known as the FIN bit) indicating that this is also the end of the transmission. The effect is that every UDP packet will be handled as if it was a new request. However, using a `nslookup` command on the client and the ND as DNS server was successful, we noticed the following pattern in the port 53 monitor:

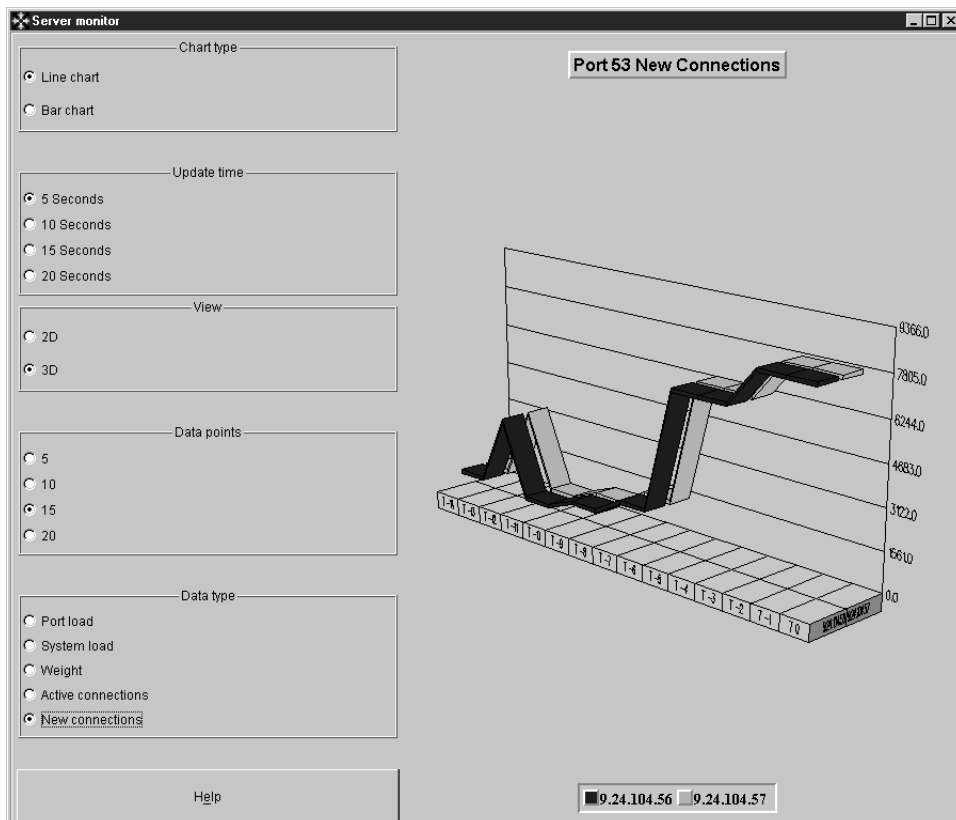


Figure 523. Monitoring New Connections on Port 53 (DNS)

This pattern appears because as each packet is also the end of a communication, the load balancing will be degraded to round-robin the packets to the available servers. Obviously the ND is capable of handling the UDP protocol, nevertheless it might be not a good idea to split up a flow of UDP packets since the receiver might wait for all of them and just receive only half of them.

As we detected some delays to DNS requests we recommend not to use the ND together with this protocol, but to stick with DNS's own capabilities to do so, that is having an internal DNS server as resolver with the two firewalls configured as forwarders. For more information about DNS read Chapter 11, "Domain Name Service (DNS)" on page 247.

When it comes to SOCKS we experienced that it simply didn't work on our first try. We enabled the SOCKS server debug option, which traces the SOCKS proxy actions, with the following steps:

- 1) Run the Registry Editor.
- 2) From the HKEY_LOCAL_MACHINE subtree, go to the following key:
\SOFTWARE\IBM\IBMFirewall\3.3\Socks\Server
- 3) Select **Edit > Add String Value**.
- 4) Add a new string with the name "Debug Log File" and set it to your desired name and path like "C:\TEMP\SOCKS.LOG"
- 5) Click **OK**.
- 6) Quit the Registry Editor.

The debug was available without any further action. We removed this registry entry as soon as we finished debugging since the log file grows quickly.

After another try to connect from the client via SOCKS we checked the created log file and found this (bracketed numbers are thread numbers):

```
[175] Accept: Dispatching thread 1 of 64...
[175] Accept: Waiting on accept or a signal
[195] Accept: Thread beginning...
[195] Check: Checking host address (00681809 == 00683596)
[195] Route: Line 61: Destination host didn't match
[195] Check: Checking host address (00681809 == 00681809)
[195] Check: Checking port range (0 <= 3562 <= 65535)?
[195] Route: Line 62: Matched
[195] Proxy: Received connection via wrong route
[195] Proxy: closing monitor handle
[195] Proxy: cleaning input io context
[195] Proxy: done cleaning up
[195] Accept: Thread exiting...
```

The mentioned line numbers correspond to those in the SOCKS configuration (counting rules and comments). Now we had a closer look into the SOCKS5.CONF configuration of firewall A and found lines 61 to 63 containing some route information:

```
[61] route 150.24.104.0/255.255.255.0 - 150.24.104.59
[62] route 9.24.104.0/255.255.255.0 - 9.24.104.56
[63] route 9.24.104.0/255.255.255.0 - 9.24.104.30
```

Obviously this seemed to be some kind of security check. Line 61 describes that incoming traffic for the non-secure network will come over 150.24.104.59 and lines 62 and 63 describe that incoming traffic for the secure network will come over 9.24.104.56 and 9.24.104.30. The rules will be checked from top to bottom and because our packets are coming from 9.24.104.30 (the cluster IP address) line 62 will deny our requests. Line 63 will never be reached.

We removed line 62 with an editor and told the SOCKS server to reload the configuration by typing

```
socks5 -config
```

on a command line. Of course this has to be done on both firewalls.

Our new try with the Web browser on the client, which was now configured to use the ND as a SOCKS server, to the Web server on the non-secure network was now successful. Your experiences may vary since the order can be different between line 62 and 63. It looks like it is dependant on the order of the installation of network adapters (try IPCONFIG on a command line).

SOCKS Config

There is actually no way to change the order of the rules with the firewall GUI but just editing the SOCKS5.CONF by hand. Since regenerating the firewall rules will also create a new SOCKS5.CONF file, we have to fix it and tell the SOCKS server to load this new configuration.

20.2.5 The Wildcard Port

If you take a closer look at Figure 520 on page 505 you will recognize that we have quite many ports. Actually you may run into the problem that ND will deny you to add more ports (don't worry: you can change the number of maximum ports on the cluster panel). But this new release of the ND has a very interesting feature called Wildcard Port that allows you to handle all traffic for a cluster IP address.

To enable this feature we removed all the present ports and added only port 0, which is an alias for the Wildcard Port. Like with all other ports we also added the both firewalls to this port and we were set. Of course having advisors for all ports doesn't make sense so we removed those also. It might be a good idea to still have a ping advisor for better control of machine availability.

For the Wildcard Port we didn't change anything on the firewall or on the client and we started testing this new environment right away. Testing any services with this new configuration was successful because the ND will redirect any incoming traffic for the cluster IP address no matter what port.

20.2.6 The Wildcard Cluster

Now we were able to load balance any service that is destined for a specific IP address (the cluster) and a specific service like HTTP proxy. But what about some other things like NAT? NAT will translate your IP addresses, but you don't point to a specific server or port.

Version 2.1 of the Network Dispatcher contains a feature called Wildcard Cluster. It is able to catch every routed packet and reroute it much like it did with the other services.

Basically we took the same scenario as described in Figure 519 on page 503 and set up the client's default gateway in the network setup panel to the cluster IP address. In addition we changed the firewalls to allow NAT traffic as described in 14.2, "Configuration of NAT" on page 341.

Now we deleted the old cluster entry in the ND's GUI and added a new one called "0.0.0.0" which is an alias for the Wildcard Cluster. We also added port 0 to this newly created cluster and added our two firewall IP addresses to this port. The definition for the cluster IP addresses is still valid on all machines. The ND is now the default gateway for the client and so it will receive every packet for IP addresses that cannot be reached by the client directly. and will redirect them to the machines defined in the cluster. The ND is not aware of any routes since it will only bypass the traffic but the destination machines should be configured to have the correct routing enabled, for example, to other intranets and maybe the default gateway to the Internet.

Again we took our client's Web browser and configured it for a direct connection not using any proxies at all. Figure 524 on page 513 shows the monitor for port 0 when we did the testing.

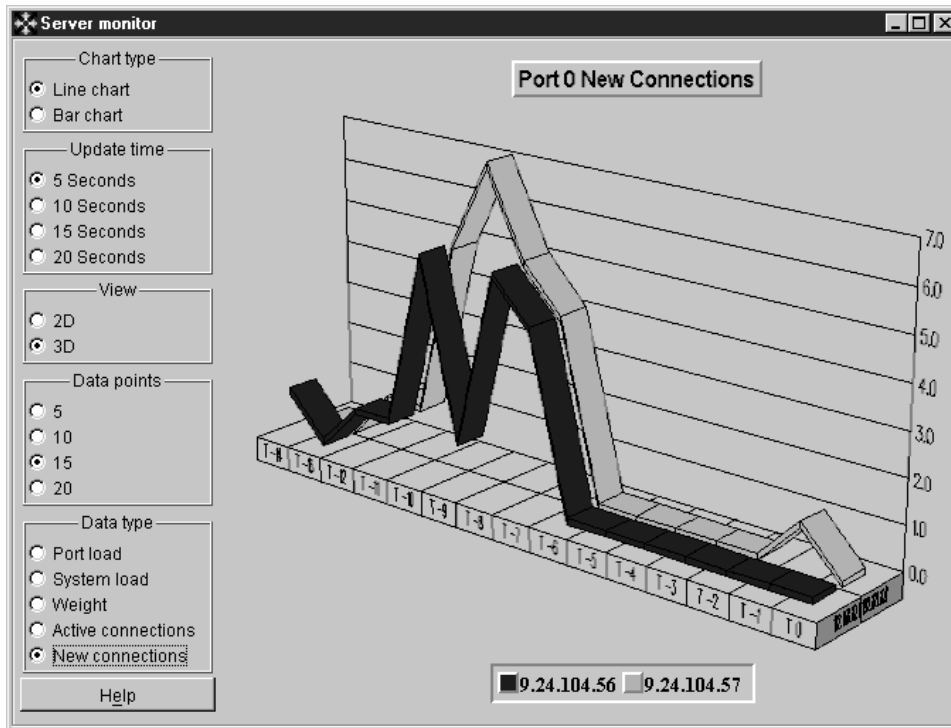


Figure 524. Monitoring New Connections on Port 0 (Wildcard Port)

As we checked the firewall log files we were able to see that both firewalls were used; the Web server's access log also showed two different IP addresses - the registered NAT addresses of firewalls A and B.

20.3 ND Advanced Usage: NAT One-to-Many Simulation

There are a lot of interesting combinations between ND and firewalls, but we came across one eye catcher that you should be aware of. The current release of IBM eNetwork Firewall for Windows NT is not capable of doing a NAT "one-to-many" mapping, that is, mapping a specific port to a specific IP address. For example, a packet that arrives to the nonsecure interface of the firewall with a destination IP address equal to the IP address of the nonsecure adapter and port 80, would be converted into the IP address of the Web server in the secure network. If another packet arrives with a destination IP address equal to the IP address of the nonsecure adapter and port 110, it will be converted to the IP address of the POP server, and so on. It is handy to have this feature because it allows you to save a lot of IP addresses. Figure 525 on page 514 shows such a scenario.

Note: Letting servers in the secure network be accessed from the Internet can be risky; it is not a safe way of allowing your customers access these services. You have to open the firewall to sessions initiated on the Internet. We present this solution in case you decide to go ahead with this approach after you have assessed the risks.

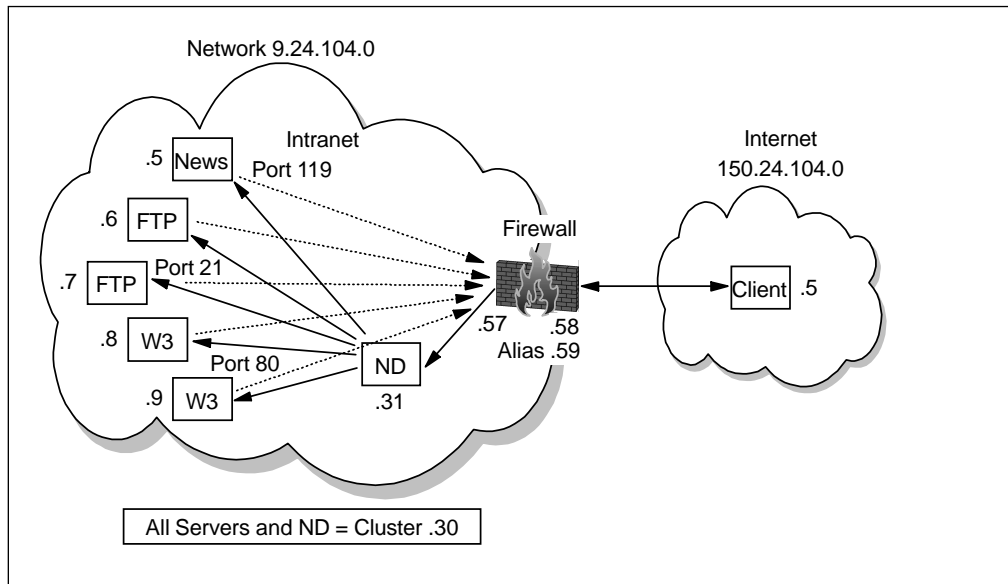


Figure 525. "One-to-Many" Simulation with ND

Configuration of this environment is straightforward:

- Configure the firewall to allow incoming traffic to the cluster IP address 9.24.104.30 and outgoing traffic for the secure network 9.24.104.0/255.255.255.0; make sure that you allow only services that you really want to get through.
- For the firewall also add a NAT mapping from 9.24.104.30 to the registered IP address 105.24.104.59.
- Add the "MS Loopback Adapter" to every server, configure it with the cluster IP address 9.24.104.30 and don't forget to get rid of the extra route to it (route delete 9.24.104.0 mask 255.255.255.0 92.24.104.30).
- Configure the ND as we did in 20.2.1, "Installation and Configuration of the ND" on page 504 and add the needed ports.

The difference now is that we don't add all servers for all ports but:

- For port 80 we add 9.24.104.8 and 9.24.104.9.
- For port 21 we add 9.24.104.6 and 9.24.104.7
- For port 119 we add only 9.24.104.5: no additional server here

If the client is trying to access Web services for IP address 150.24.104.59, it will end up on server 9.24.104.8 or 9.24.104.9. For using FTP it will reach 9.24.104.6 or 9.24.104.7 and trying to use the News server it will be served by 9.24.104.5 only.

We reach this goal because the firewall will map the non-secure IP address 150.24.104.59 to 9.24.104.30. Now the ND is able to receive the packets and will forward them depending on the defined ports and servers. Now you are able to hide many machines behind one IP address.

20.4 High Availability

Another feature of the Network Dispatcher is the High-Availability (HA) option. See Figure 519 on page 503; we have two firewalls, but the ND is still a single point of failure as already mentioned at the end of section 20.1, "The Basics of ND" on page 503. To avoid this we have to set up another ND machine and configure both NDs with the HA option, so that if one fails, the other will take over. It is necessary for the ND to check if the firewalls are alive so we have to configure the rules to allow a ping or the specific advisor ports.

Now that we have this side highly available, how about the non-secure side? Of course it would be necessary to have a transparent flow also from the Internet to talk to just one IP address. The solution would be to also set up two NDs with the HA option configured on the nonsecure side.

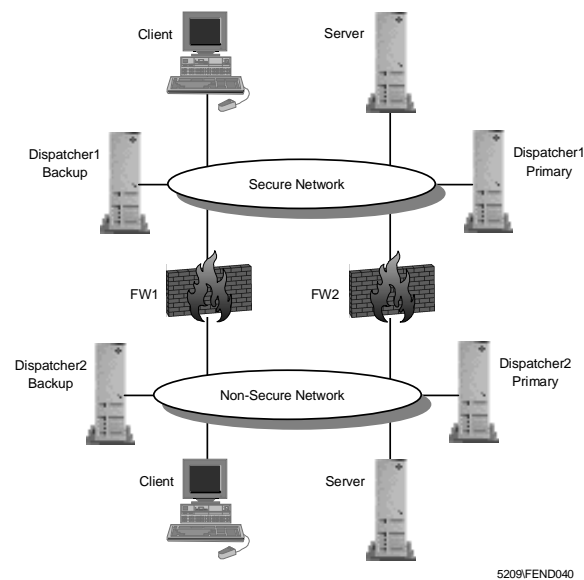


Figure 526. High Availability Scenario

To build such a scenario you would end up with at least six machines and you would also have to set up a DMZ with a screening router to have basic protection for the non-secure NDs. Also the configuration will get complicated because you have to add another "MS Loopback Adapter" on each firewall.

In order to simplify the firewall high availability configuration, the best solution would be to co-locate the IBM eNetwork Firewall for Windows NT and the Network Dispatcher in the same machine. This allows you to work with only two machines. We did not set up this configuration.

The AIX versions of the firewall and the Network Dispatcher are able to co-locate the firewall and the ND on one machine, Both machines run the firewall, the inside and the outside cluster, the HA option and, on top of that, they also could do load balancing. This configuration has been tested and works fine. See the soon to be published redbook on WebSphere Performance Pack V2.0 for a detailed description on how to implement this solution.

Chapter 21. Integration with Other Products

In this chapter we look at several products related to the IBM eNetwork Firewall for NT.

Warning

The level of the products mentioned in this chapter is as of March 1998. Some of them have released new versions since then, but the basic functions that we describe remain the same.

21.1 IBM DB2

The IBM Firewall allows you to log denied or permitted IP traffic, Socks and proxy usage, and other events that happen during day-to-day usage. You can use IBM DB2 to extract information out of these log files. This includes information about attack rates, type of attacks, and information of resource usage.

With the IBM Firewall Report Utilities it is possible to convert the firewall log files into import files for DB2 (see 16.4, "Building Reports" on page 436). In this way you can use all the power of the Structured Query Language (SQL), or other tools such as IBM's Visualizer or Query Management Facility to query the data and generate reports.

In this section we describe how to use DB2 to create tables for the log data, process the import files, and generate sample reports.

You find all SQL scripts mentioned in this section in the following directory
C:\Program Files\IBM\Firewall\sample\report\.

Table 46. IBM Firewall SQL Scripts

File	Description
fwschema.ddl	File of SQL Data Definition Language (DDL) statements, suitable for defining the database tables.
fwimport.dat	File of DB2 import statements, suitable for importing the DEL files into the database tables.
fwqrysmpl.dml	File of SQL Data Manipulation Language (DML) statements, suitable for generating sample reports.

21.1.1 Create the Initial Database

We assume that you have DB2 installed on a machine and have an instance defined. Execute the db2cmd command to open a DB2 Command window.

First you have to create an empty database. We create the database fwlog with the following command:

```
C:\>db2 create database fwlog
DB20000I The CREATE DATABASE command completed successfully.

C:\>
```

Figure 527. Create the Initial Database

Before you can work with a database, you have to connect to it. We connect to the database fwlog by:

```
C:\>db2 connect to fwlog

      Database Connection Information

Database product      = DB2/NT 5.0.0
SQL authorization ID = DB2ADMIN
Local database alias = FWLOG

C:\>
```

Figure 528. Connect to Database

Similarly you have to disconnect from the database after usage. Disconnect with the following command:

```
C:\>db2 disconnect fwlog
DB20000I The SQL DISCONNECT command completed successfully.

C:\>
```

Figure 529. Disconnect from Database

Once connected to the database fwlog, you need to create the tables and indexes used for the log data. The following command must be issued once:

```
C:\>db2 -vf fwschema.ddl > schema.out

C:\>
```

Figure 530. Creating Tables and Indexes

Check schema.out for possible errors.


```
C:\>type schema.out
CREATE TABLE ADMIN_ALERT( DATE_TIME    TIMESTAMP    NOT NULL,
                           FIREWALL    VARCHAR(100) NOT NULL,
                           PID          INT          NOT NULL,
                           MSG_NUM     SMALLINT    NOT NULL,
                           USERID      CHAR(8),
                           ACTION      CHAR(7),
                           NUM_COUNT   SMALLINT,
                           TAG_MSG_NUM CHAR(8),
                           SRC_IP      CHAR(15),
                           DST_IP      CHAR(15),
                           AUTH_METHOD CHAR(20),
                           NETWORK     CHAR(25),
                           HOST_NAME   VARCHAR(100),
                           TIMEOUT_SEC SMALLINT,
                           CONN_USERID CHAR(8),
                           APPLICATION CHAR(10),
                           ERROR_NUM   SMALLINT,
                           FILE_NAME   VARCHAR(100),
                           LOG_FILE    CHAR(25)    NOT NULL,
                           LOG_LINE_NUM INT        NOT NULL,
                           REPEAT_NUM  SMALLINT    NOT NULL,
                           LOG_LEVEL   CHAR(1))

DB20000I The SQL command completed successfully.

CREATE UNIQUE INDEX UIX_ADMIN_ALERT ON ADMIN_ALERT (LOG_FILE, LOG_LINE_NUM, REPEAT_NUM)
DB20000I The SQL command completed successfully.

...

C:\>
```

Figure 531. Verifying Creation of Tables and Indexes

21.1.2 Import the Log Data

Change to the directory where you put all the import files that were generated from the log files. Run the following command to import the data from the files into the DB2 database tables:

```
C:\Temp\log>db2 -vf fwimport.dat > import.out

C:\Temp\log>
```

Figure 532. Import the Log Data

If the user of the import command is not the creator of the tables, the table names in the fwimport.dat file may need to be prefixed with the name of the table creator and a dot. Each import command in the fwimport.dat file produces information in a <tablename>.msg file. Furthermore, there is a summary of the records processed on the standard out: import.out.

```

C:\Temp\log>type import.out
IMPORT FROM a_alert.tbl OF DEL MODIFIED BY COLDEL;
      MESSAGES a_alert.msg INSERT INTO ADMIN_ALERT

Number of rows read      = 296
Number of rows skipped  = 0
Number of rows inserted = 296
Number of rows updated  = 0
Number of rows rejected = 0
Number of rows committed = 296

SQL3107W There is at least one warning message in the message file.

...

C:\Temp\log>

```

Figure 533. Verifying Importing of Log Data

21.1.3 Sample Queries

The file `fwqrysmp.dml`, provided with the IBM Firewall Report Utilities, contains some SQL sample queries you can try on your database. The following command starts the query:

```

C:\Temp\log>db2 -vf fwqrysmp.dml > report.out

```

Figure 534. Running the Sample Queries

The results can be found in `report.out`. To illustrate the capabilities of SQL we present you now with a few sample queries from `fwqrysmp.dml`. The query in Figure 535 shows sessions opened by a specific user ID.

```

SELECT USERID, APPLICATION, DATE_TIME
from SESSION
  where USERID = 'sven'
     and SESSION_EVENT = 'begin'

USERID  APPLICATION  DATE_TIME
-----
sven    FTP          1998-02-13-18.42.00.000000
sven    FTP          1998-02-13-18.44.00.000000
sven    TELNET       1998-02-13-18.47.00.000000
sven    TELNET       1998-02-13-18.48.00.000000

4 record(s) selected.

```

Figure 535. Sample Query 1

The next query gets the amount of data transferred for these sessions.

```

SELECT USERID, APPLICATION, BYTES, DATE_TIME
from SESSION
  where USERID = 'sven'
     and SESSION_EVENT = 'end'

```

USERID	APPLICATION	BYTES	DATE_TIME
sven	ftp	23484	1998-02-13-18.44.00.000000
sven	ftp	3803	1998-02-13-18.45.00.000000

2 record(s) selected.

Figure 536. Sample Query 2

The last query shows all instances where a deny filter rule was applied in a given time period.

```

SELECT DATE_TIME, DIRECTION, SRC_IP, DST_IP, PROTOCOL, SRC_PORT, DST_PORT, INTERFACE
from FILTER_MATCH
  where ACTION='deny'
     and DATE_TIME between '1998-02-24-00.00.00' and '1998-02-24-11.59.59'

```

DATE_TIME	DIRECTION	SRC_IP	DST_IP	PROTOCOL	SRC_PORT	DST_PORT	INTERFACE
1998-02-24-08.32.00.000000	inbound	192.168.51.4	9.24.104.70	icmp	8	0	nonsecure
1998-02-24-09.44.00.000000	inbound	9.24.104.70	9.24.104.191	udp	2220	53	secure
1998-02-24-09.01.00.000000	inbound	192.168.51.4	150.53.104.56	tcp	1025	23	nonsecure
1998-02-24-10.23.00.000000	inbound	192.168.51.4	150.53.104.56	tcp	1450	21	nonsecure

4 record(s) selected.

Figure 537. Sample Query 3

Which queries you need depends on the kind of reports you want to make. We suggest to study the example queries and the file `fwschema.dll` which contains the structure of the tables. After that it will be relatively easy to build your own queries.

21.2 Aventail AutoSOCKS

Aventail AutoSOCKS is a 16 and 32-bit Windows application used to enable other network applications for firewall traversal and virtual private networks. It enables any TCP/IP application to traverse existing SOCKS V4 or V5-based firewalls.

AutoSOCKS makes Windows-based TCP/IP applications SOCKS-compliant. By implementing AutoSOCKS with a corporation's existing TCP/IP applications, clients can communicate through the SOCKS-enabled firewalls.

A diagram of the flow for TCP/IP services in a Windows client is shown in Figure 538 on page 522.

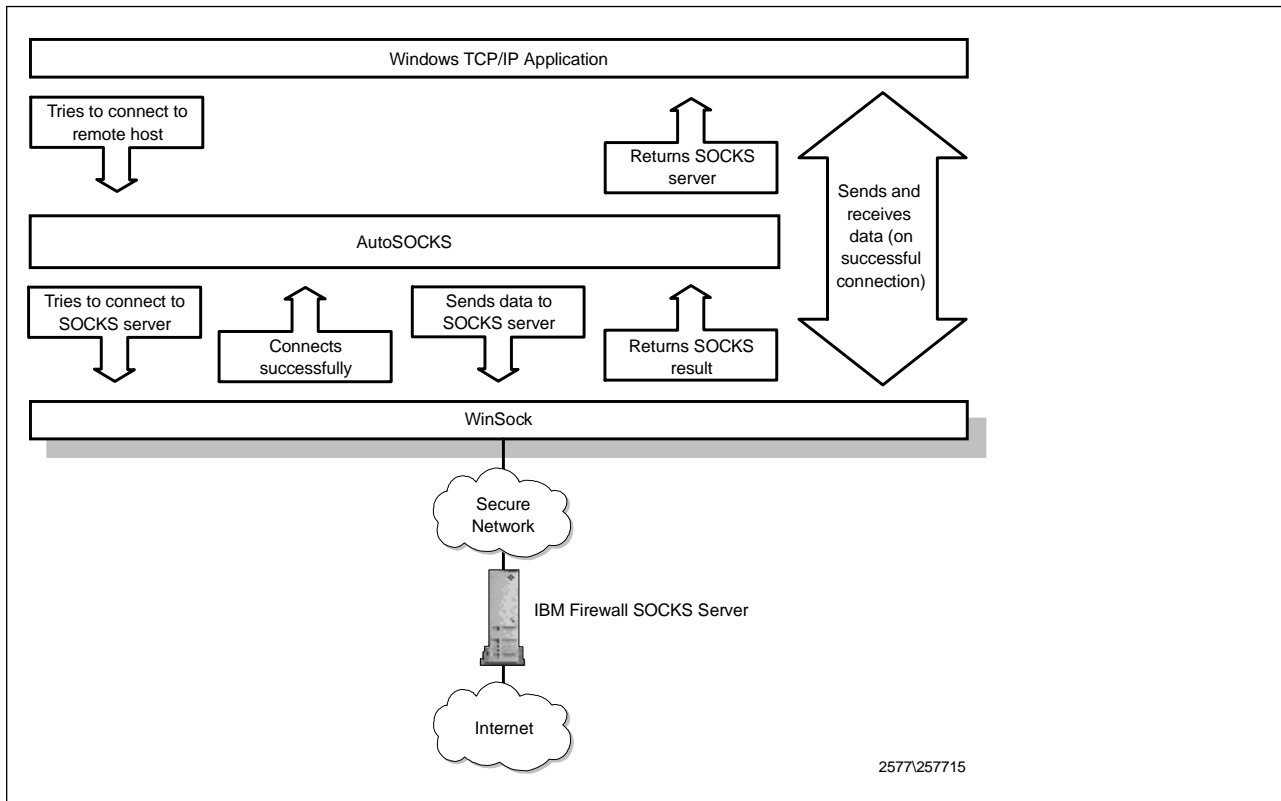


Figure 538. Aventail AutoSOCKS Flow

AutoSOCKS works by intercepting WinSock communication requests issued by applications and then processes the requests based on a set of rules. These rules govern whether or not a WinSock request is redirected through a SOCKS server. While communications within the local network may proceed unchanged, all communication with external networks can be redirected through one or many SOCKS servers.

21.2.1 Major Features

The following is a list of AutoSOCKS features provided by the vendor:

- Supports all Windows platforms
 - Windows 3.x, Windows for Workgroups 3.11, Windows 95 and Windows NT
- Supports multiple authentication and encryption standards
 - Simple username/password
 - Challenge-Handshake Authentication Protocol (CHAP)
 - Challenge-Response Authentication Method (CRAM)
 - Kerberos and SSL (optional)
- End-user transparency
 - Integrates with existing desktop applications and TCP/IP stacks
 - Seamlessly routes connections from Windows applications to external networks through a SOCKS server
 - Transparently negotiates authentication and encryption with SOCKS V5 server

- Support for multiple SOCKS servers
- No modification to Windows system components and environment
- Standards support
 - Interoperable with 16- and 32-bit WinSock 1.1 applications
 - Supports all Windows, Windows 95 and Windows NT TCP/IP stacks
 - Supports publicly available SOCKS V4 and V5 standards
- Network administration
 - User interface provides a single configuration point for all WinSock client applications
 - Online documentation and help
 - Logging tool for troubleshooting
 - Ping and traceroute for monitoring network connections

21.2.2 Installing Aventail AutoSOCKS

The AutoSOCKS package comes as an InstallShield self-extracting file, so setting it up is easy. Run the installation program from the file explorer or from the command prompt.



Figure 539. Installing AutoSOCKS

The installation gives you the option to start AutoSOCKS automatically. Thus, every time you boot your computer, all your TCP/IP applications will automatically be SOCKSified.

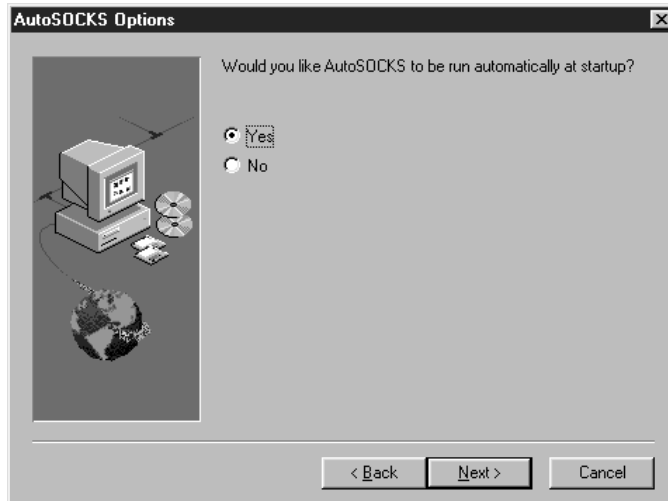


Figure 540. Option to start AutoSOCKS Automatically



Figure 541. Program Location

After the installation, you will have a program group for AutoSOCKS.



Figure 542. The AutoSOCKS Program Group

21.2.3 Configuring AutoSOCKS

AutoSOCKS comes with a Configuration Wizard that you can use to set up the first time around. The Wizard asks a series of questions to configure your system to use an existing SOCKS server on the network, in this case, the firewall.



Figure 543. The AutoSOCKS Configuration Wizard

The Wizard creates a configuration file based on your responses.

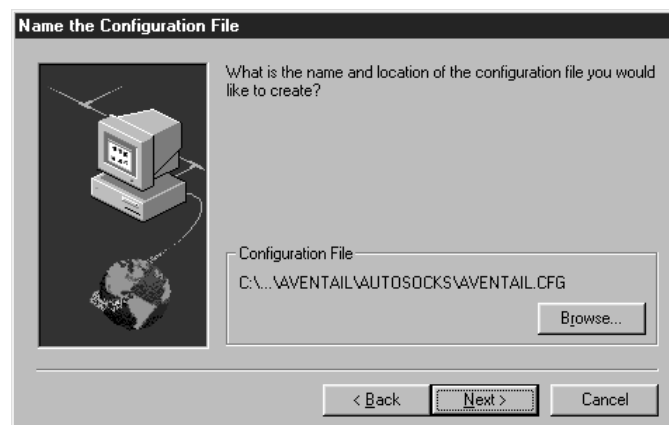


Figure 544. The Configuration File Location

You can use this configuration file on other installations of AutoSOCKS so you do not have to run the Wizard every time.

Next, you will be asked to specify the SOCKS server and its type.

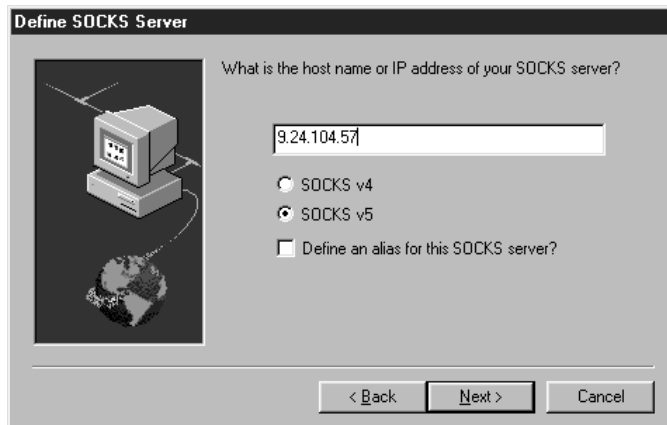


Figure 545. Specifying the SOCKS server

The Wizard will ask you if your configuration will be used to proxy traffic to the Internet or to a private network.

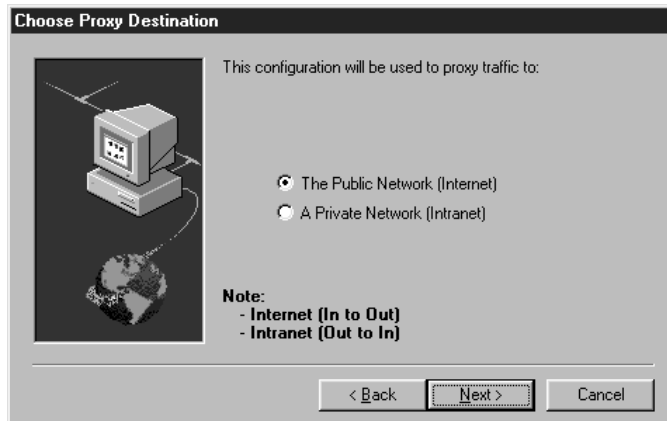


Figure 546. Specifying the Direction of Traffic

If you select the option to proxy traffic to the Internet, the Wizard will ask you what your internal network address and subnet mask are. Traffic within this network will not use the proxy.

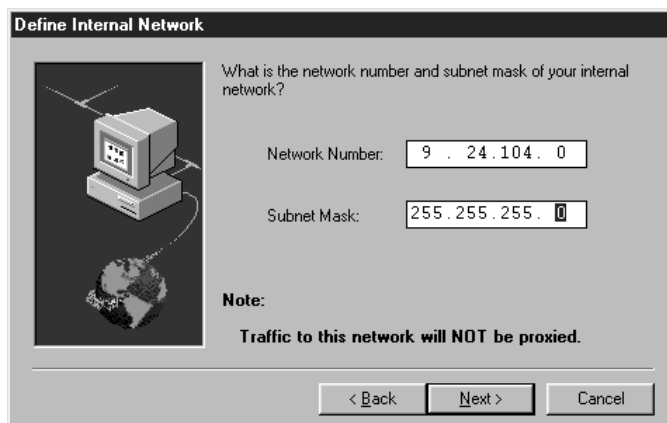


Figure 547. Specifying the Internal Network Address

The Wizard will ask you to specify the internal domain name.

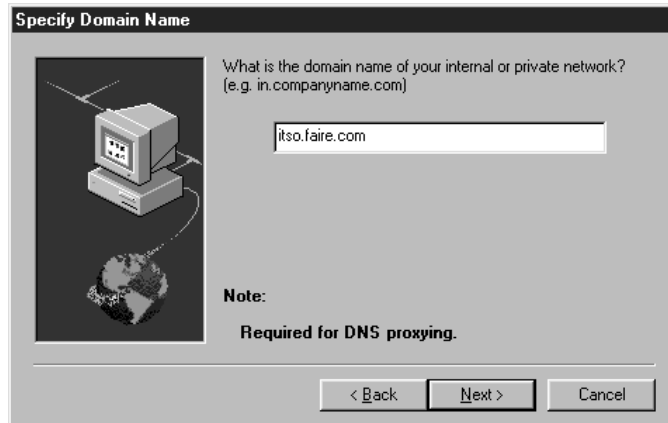


Figure 548. Specifying the Internal Domain Name

Then comes the confirmation screen.

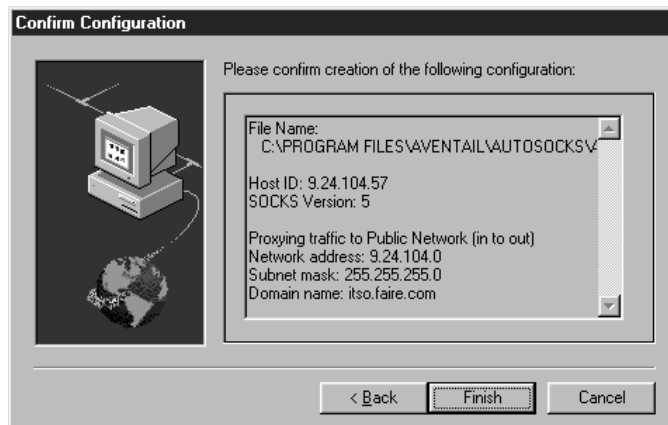


Figure 549. Confirmation

The Wizard gives you the option to make the configuration file you created as the default.

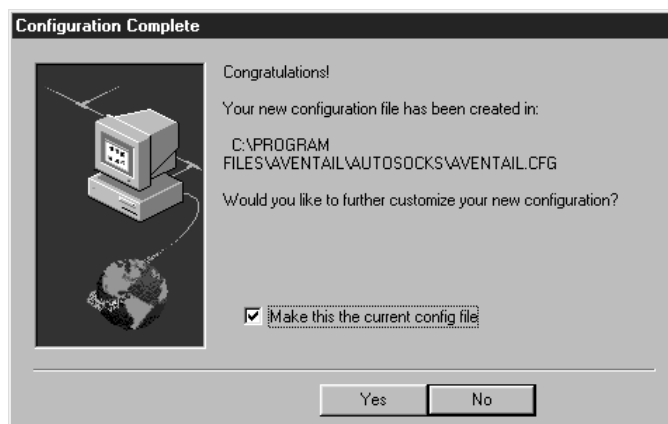


Figure 550. Making the Configuration File the Default

Now you're done.

Before you start AutoSOCKS, make sure that you have the license file in the program's directory. The license file should be part of the install media from Aventail.

You can either reboot the system or run AutoSOCKS from the program group. This will SOCKSify your Winsock stack.

21.2.4 AutoSOCKS Utilities

AutoSOCKS comes with utilities that you can use to troubleshoot the SOCKS installation on your client.

21.2.4.1 Ping

Use S5 Ping to test connectivity from your AutoSOCKS client to a target system on the other side of the firewall. You can do both ping and traceroute.

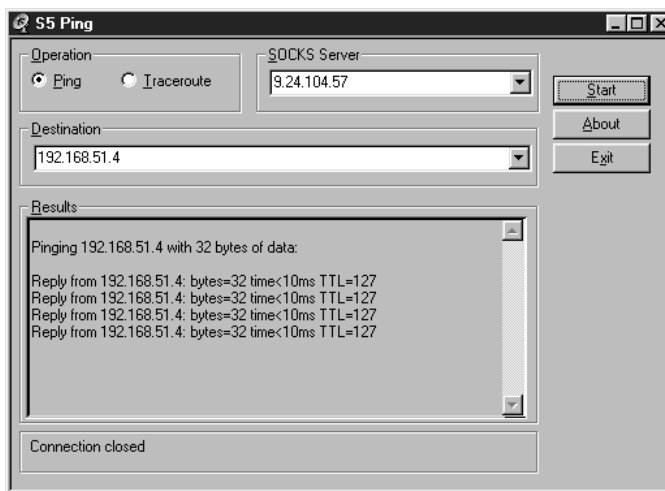


Figure 551. S5 Ping at Work

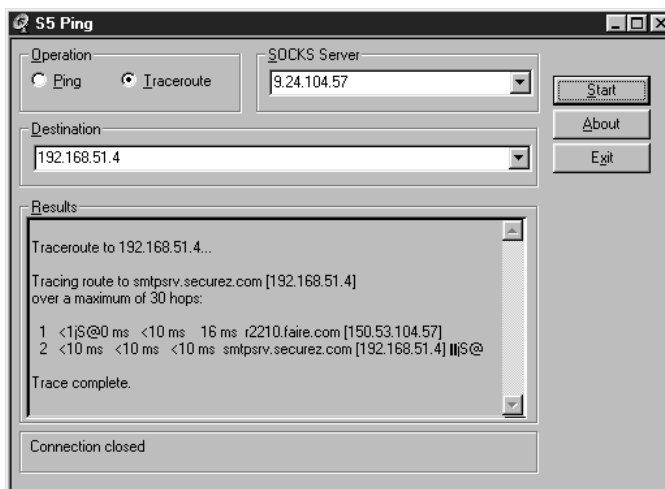


Figure 552. S5 Ping Doing a Traceroute

21.2.4.2 Logging Tool

The AutoSOCKS logging tools shows you all the SOCKS traffic being generated by your client. Use it to resolve connection or transaction problems with the SOCKS server.

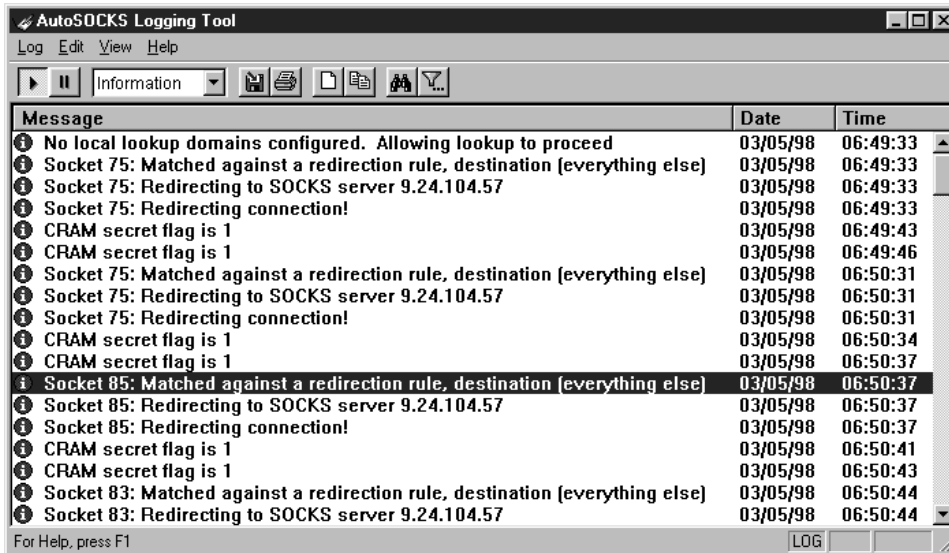


Figure 553. Logging Tool

21.3 IBM Web Traffic Express

IBM Web Traffic Express is a caching proxy server with many additional features, for example:

- It uses an innovative caching scheme, just released from IBM's research laboratories, to improve end-user response time.
- It allows an administrator to control automatic cache refresh options and whether certain pages are always loaded from cache.
- It is capable of supporting cache sizes up to 600 terabytes (depending on available hardware).
- It implements *Platform for Internet Content Selection* (PICS)-based filtering at the server.
- It is HTTP 1.1 compatible, including support of persistent sessions.

Please visit the IBM Web Traffic Express Web site at <http://www.ics.raleigh.ibm.com/WebTrafficExpress/> for a full description of all features and a complete configuration description.

This section describes two integration examples of the IBM Web Traffic Express HTTP proxy server and the IBM Firewall. In the first example both servers are installed on the same host; the second example shows a scenario with two separate hosts.

21.3.1 Using IBM Web Traffic Express on Firewall

We use the scenario shown in Figure 554 for using IBM Web Traffic Express on the firewall.

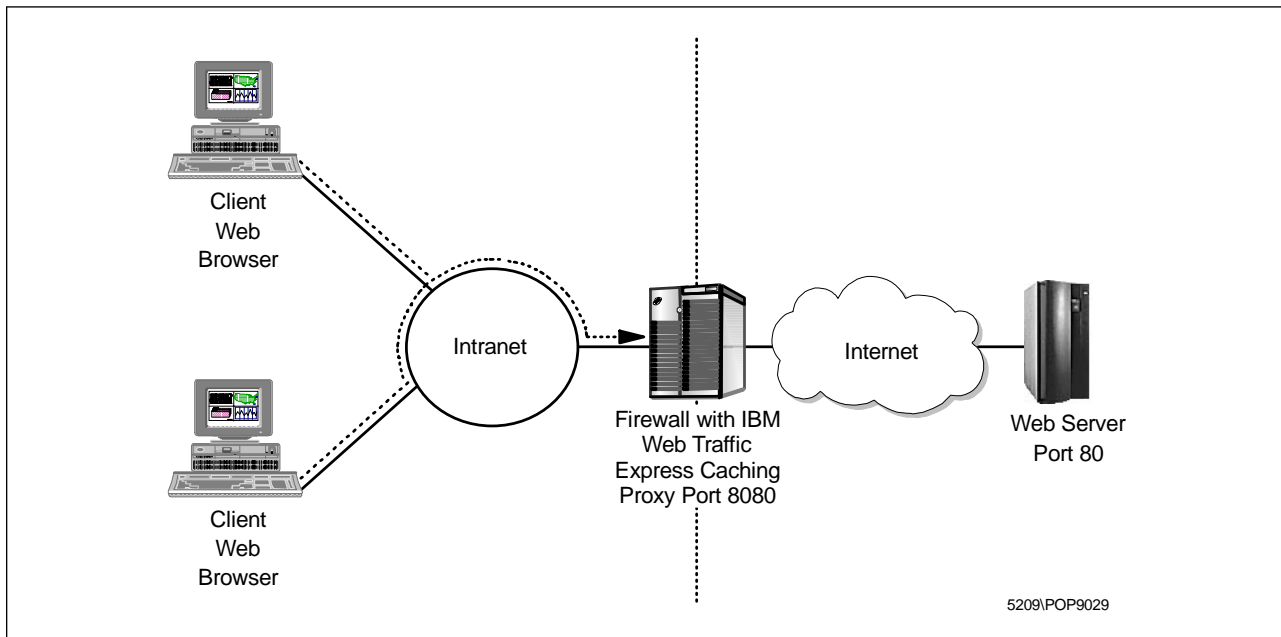


Figure 554. Scenario 1: IBM Web Traffic Express on the Firewall

You have to do the following steps for installation:

1. Disable the firewall HTTP proxy server.
2. Install IBM Web Traffic Express on the firewall.
3. Do the basic proxy configuration.
4. Define connections for remote administration.
5. Define connections for users.
6. Point your Web browser to the HTTP proxy server.
7. Further proxy configuration.

Be sure to disconnect the firewall from the non-secure network during installation to prevent security problems.

21.3.1.1 Disable Firewall HTTP Proxy

You get an HTTP proxy server with the IBM Firewall, but it has fewer features than the IBM Web Traffic Express proxy server. So you might want to exchange the built-in proxy with IBM Web Traffic Express. Both proxy servers are integrated in the Windows NT environment as services. The IBM Web Traffic Express will replace the functions of the IBM Firewall HTTP proxy server. Therefore, you need to stop and disable the IBM Firewall HTTP proxy server from the Services Control Panel in Windows NT.

To open the Services Control Panel select the **Start** button from the Windows NT task bar. From the following menu select **Settings** and then **Control Panel**. Now a window with all control panels appears. Double-click the **Services** icon to open

the Windows NT Services Control Panel. From this window (see Figure 555 on page 531) you can stop and disable the HTTP proxy server.

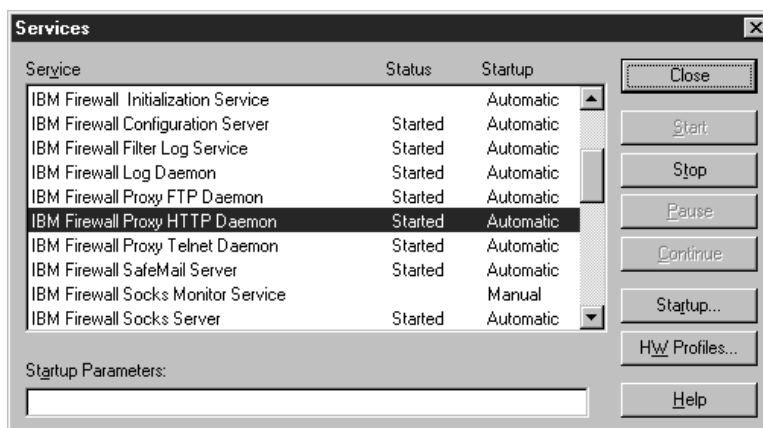


Figure 555. Windows NT Services

To stop the HTTP proxy server, select the **IBM Firewall Proxy HTTP Daemon** service from the list and click the **Stop** button.

To prevent this service from starting automatically at system reboot you need to disable the service. Therefore select the service again and click the **Startup** button from the Services Control Panel in Figure 555.



Figure 556. Disable IBM Firewall Proxy HTTP Daemon

Select the **Disabled** radio button and click the **OK** button to close the window. Your proxy server should now be disabled as in Figure 557 on page 532.

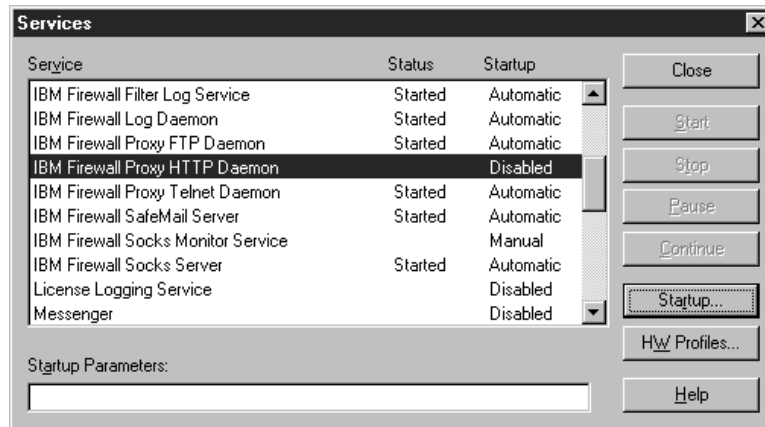


Figure 557. Disabled Firewall HTTP Proxy

21.3.1.2 Install IBM Web Traffic Express

Installation of the IBM Web Traffic Express is very straightforward. Log on with a user ID that is a member of the administrators group. Simply run setup.exe from the installation source directory. You will see the following window.



Figure 558. IBM Web Traffic Express Welcome Window

Click on **Next** to come to the component selection window.

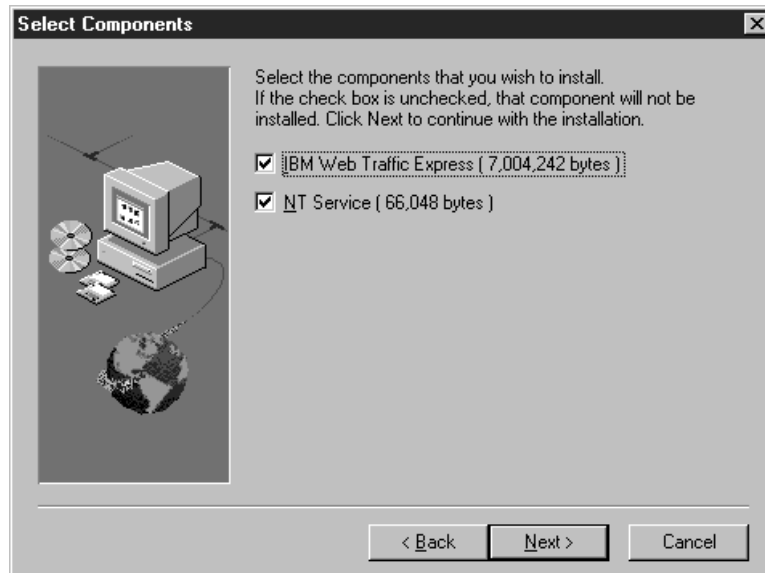


Figure 559. Select Components to Install

If you install IBM Web Traffic Express as a Windows NT service, you can make the server start automatically at system reboot. So use the default settings and accept both components by clicking the **Next** button. You are now at the target directory selection window.



Figure 560. Target Directory Selection

You can accept the proposed destination directory or choose a new one with the **Browse** button. Unfortunately Setup won't accept any path containing a space, such as C:\Program Files\IBM\WTE\. Therefore we accepted the default directory. Click the **Next** button again to get to the next window.

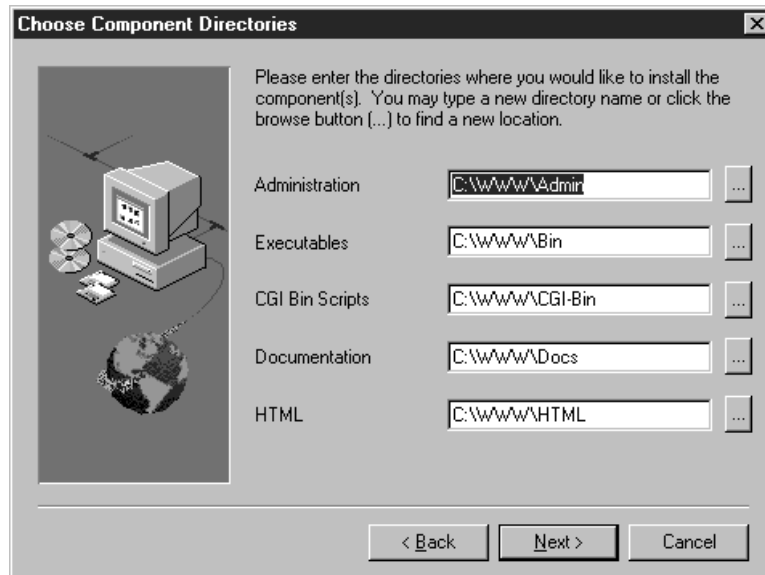


Figure 561. Component Directory Selection 1

This and the next window allows you to select directories for all components of the HTTP proxy. We recommend to accept the default values by clicking **Next**. This creates sub-directories in your destination directory.

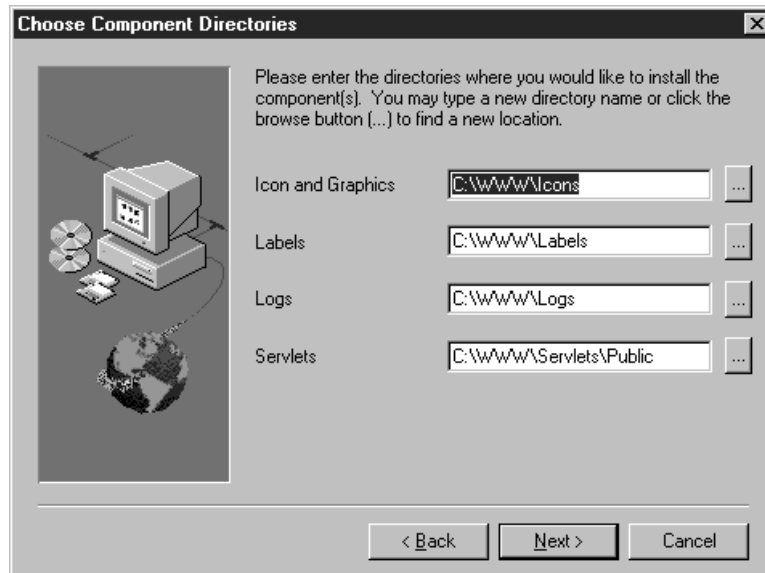


Figure 562. Component Directory Selection 2

Simply click the **Next** button. This will take you to the Configuration Parameters window.

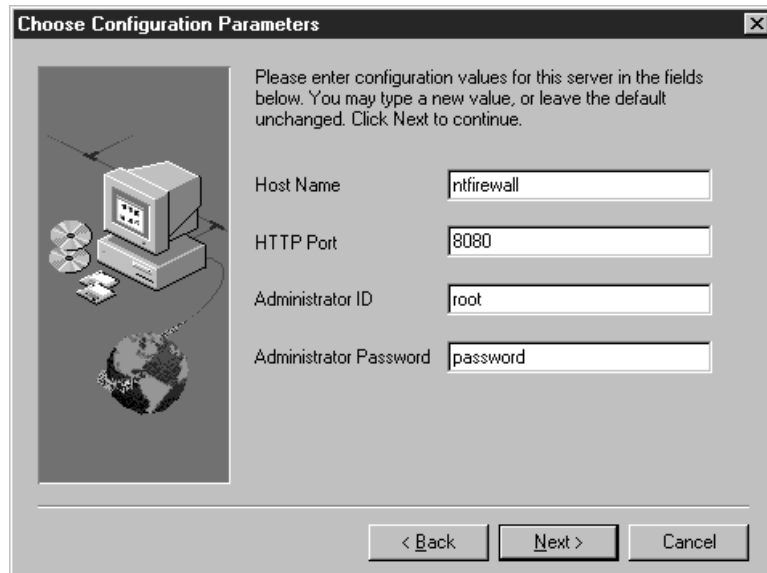


Figure 563. Host and Administrator Configuration

The host name should be filled in automatically. In our case it's ntfirewall (9.24.104.57). Fill in a proper HTTP port such as 8080. We recommend you not to use 80, because 80 is widely used for content Web servers.

Admin Port and User Port

The IBM IBM Web Traffic Express does not support different ports for the administration and proxy usage, but this doesn't mean that everyone can access the administration pages. They are password protected and you can restrict access to certain IP addresses.

Choose your administrator ID and the corresponding password. This need not be the same as the local login account because you also can configure IBM Web Traffic Express remotely. Click **Next** to continue.

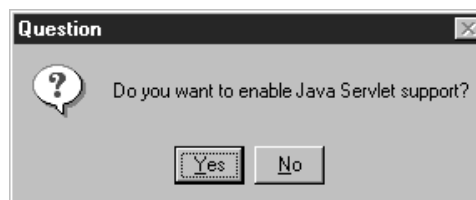


Figure 564. Java Servlet Support

Answer the Java servlet question with **Yes**. Now the installation starts. If the IBM Web Traffic Express is installed successfully, you get the window in Figure 565 on page 536.

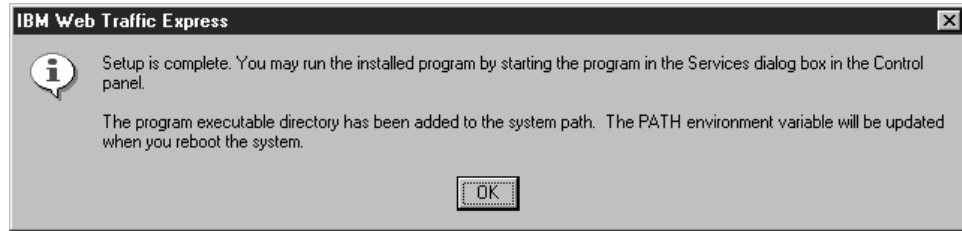


Figure 565. Finish Installation

Click **OK** to finish installation but don't reboot your computer now. You need first to enable the newly installed proxy with the Windows NT Services Control Panel.

Open the Services Control Panel and select the **IBM Web Traffic Express** service. To start this service automatically at system reboot click the **Startup** button and select the **Automatic** radio button from the following window. Click the **OK** button to close the window. Your proxy server should now be enabled as in Figure 566.

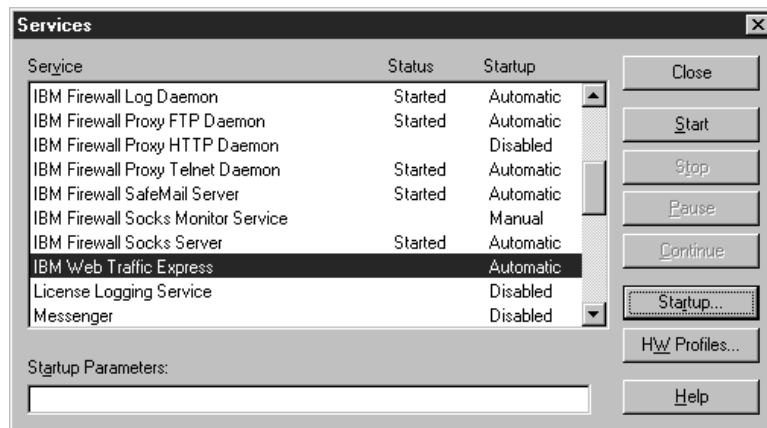


Figure 566. Enabled IBM Web Traffic Express Service

Reboot your computer to activate changes on the PATH variable (see Figure 565).

21.3.1.3 Basic Configuration

Please be sure that the IBM Web Traffic Express operates as a proxy server only. It also can operate as a content Web server, but for higher security it should not provide public Web pages from your firewall. To check this, start your Web browser and open the URL for Web-based configuration. In our case it's <http://9.24.104.57:8080/>. As we have defined port 8080 as our proxy port we added it to the URL. Your browser shows now the IBM Web Traffic Express main page.

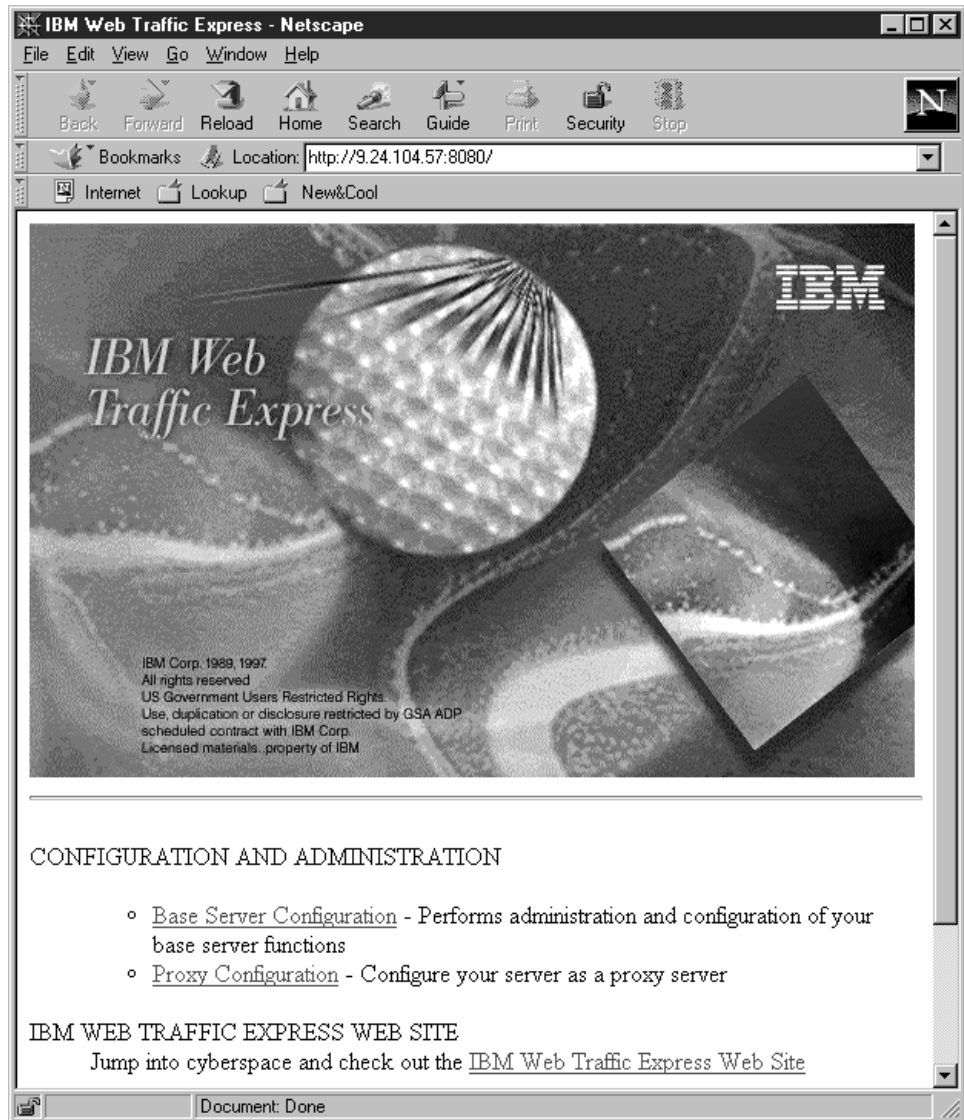


Figure 567. IBM Web Traffic Express Main Page

From this page you can administer the proxy server, view the documentation and visit the IBM Web Traffic Express Web site by selecting **Proxy Configuration**. You now will be asked for your user ID and password. Enter the appropriate data and select **OK**.



Figure 568. IBM Web Traffic Express Proxy Authentication

You get the Proxy Configuration and Administration Forms page shown in Figure 569 on page 538.

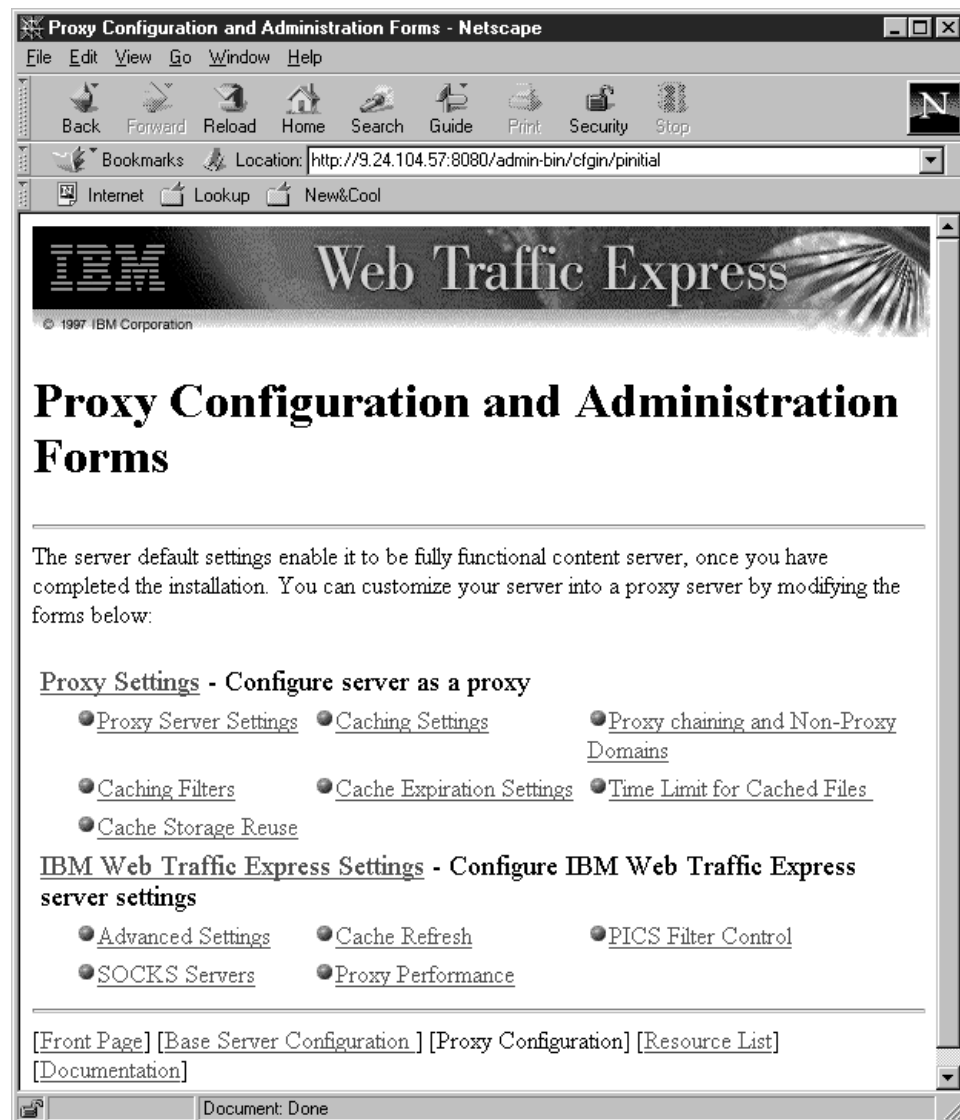


Figure 569. IBM Web Traffic Express Proxy Configuration and Administration Forms

Select **Proxy Performance** from this window.

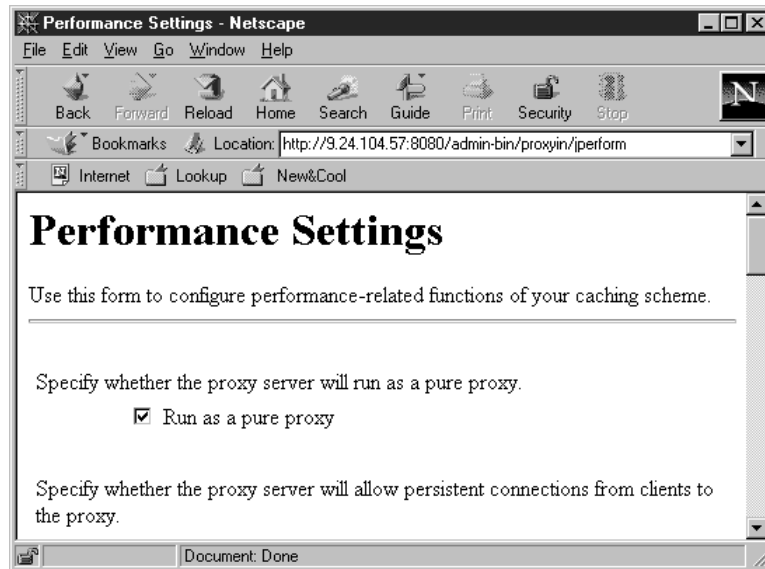


Figure 570. Pure Proxy Configuration

Make sure that **Run as a pure proxy** is checked.

Please refer to the *IBM Web Traffic Express for Multiplatforms User's Guide* for a full configuration of the IBM Web Traffic Express HTTP proxy.

21.3.1.4 Connections for Remote Configuration

To allow remote configuration for the IBM Web Traffic Express you must define a connection between your local host and the firewall with the IBM Firewall Configuration Client. If you have already defined a connection for your internal network to use the firewall HTTP proxy and your host is in the same network, you don't need to add a new connection. Please make sure that the connection is defined for the appropriate ports. The default is 8080.

Figure 571 on page 540 shows the required connection to administer the IBM Web Traffic Express remotely.

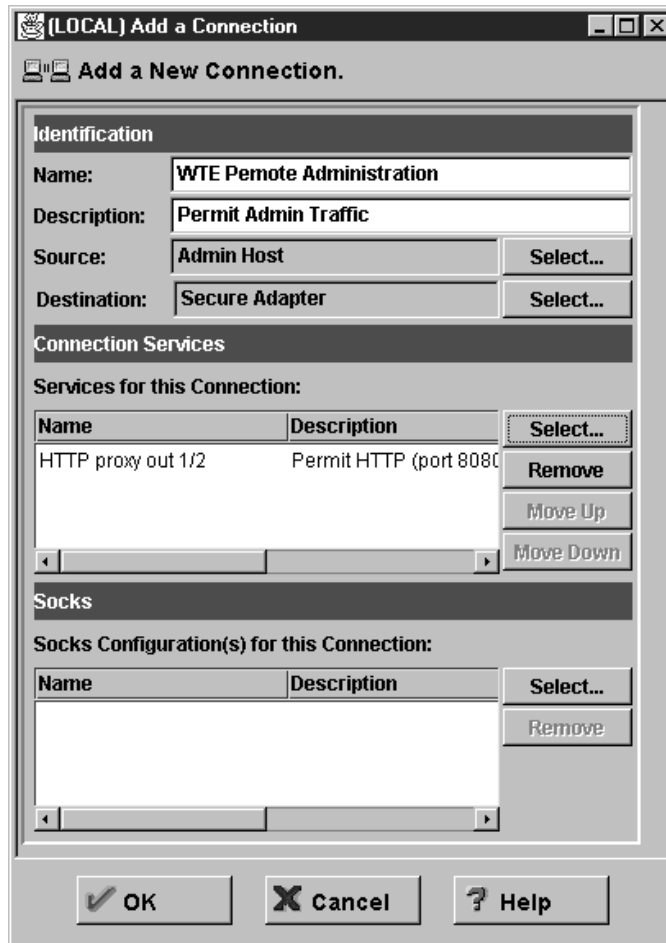


Figure 571. Remote Configuration Connection

Please refer to 13.6, "Using the HTTP Proxy" on page 316 for details.

21.3.1.5 Connections for Users

You must define two connections at the firewall to allow your users to use the IBM Web Traffic Express for outbound connections. These connections are the same as for the IBM Firewall HTTP proxy defined in 13.6.2.1, "Connections" on page 319. We repeat them here shortly.

Figure 572 on page 541 and Figure 573 on page 542 show the HTTP proxy connections required in the secure and non-secure network respectively.



Figure 572. HTTP Proxy Connection on Secure Network

For the connection from the secure hosts to the firewall you only need one service because other protocols supported though the HTTP proxy are tunneled through HTTP.

But for the outside connection you have to specify all desired protocols. In this example we have specified FTP, HTTP, and HTTPS.

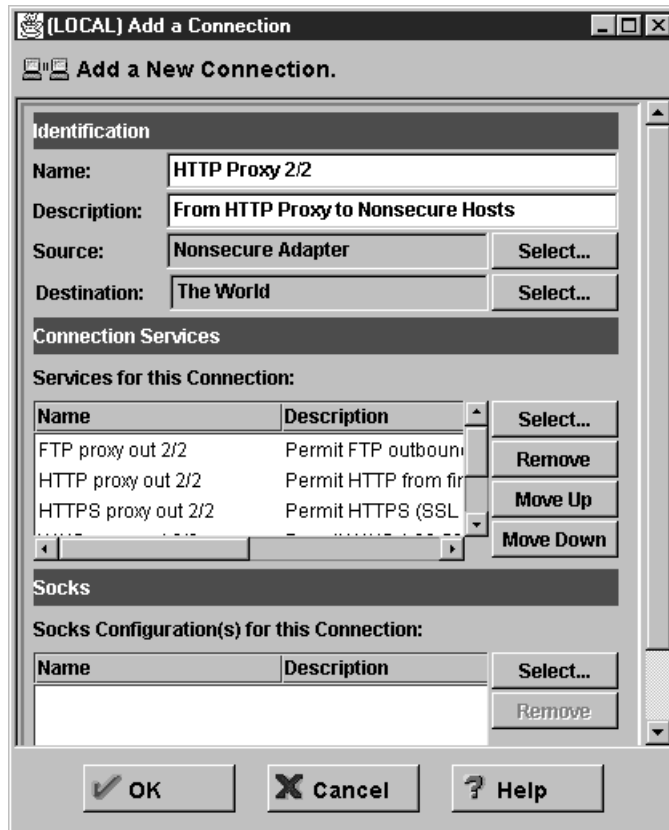


Figure 573. HTTP Proxy Connection on Non-secure Network

21.3.1.6 Browser Configuration

The last step is to point your browser to the IBM Web Traffic Express at the firewall with the appropriate port.

To open the Preferences window: from Netscape Navigator Version 4, select **Edit** from the menu and then **Preferences**. From the Category navigation tree select **Advanced** and click on **Proxies**. Figure 574 on page 543 shows the Preferences window from the Navigator.

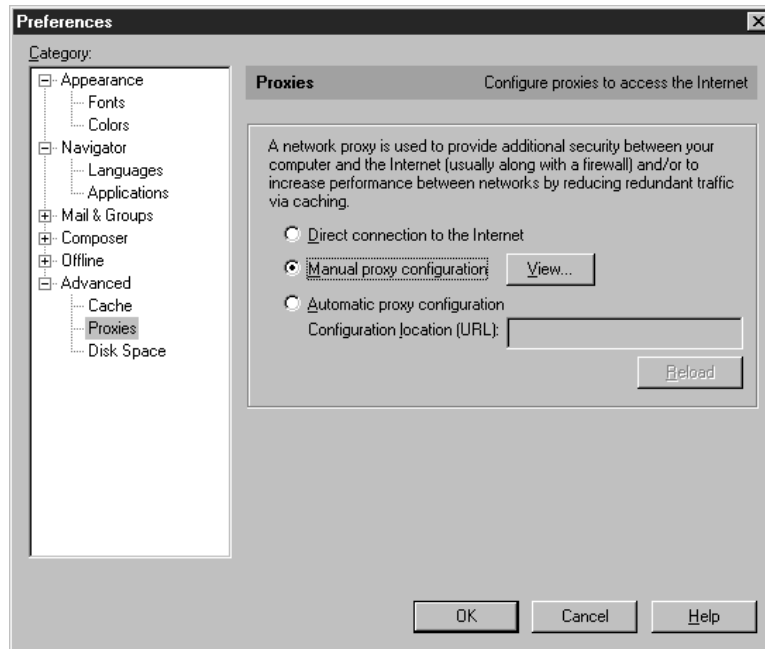


Figure 574. Netscape Navigator Proxy Preferences Window

Now switch the radio button to **Manual proxy configuration** and click on the **View** button right from this choice. You should now see the Manual Proxy Configuration window. Figure 575 shows our configuration settings.

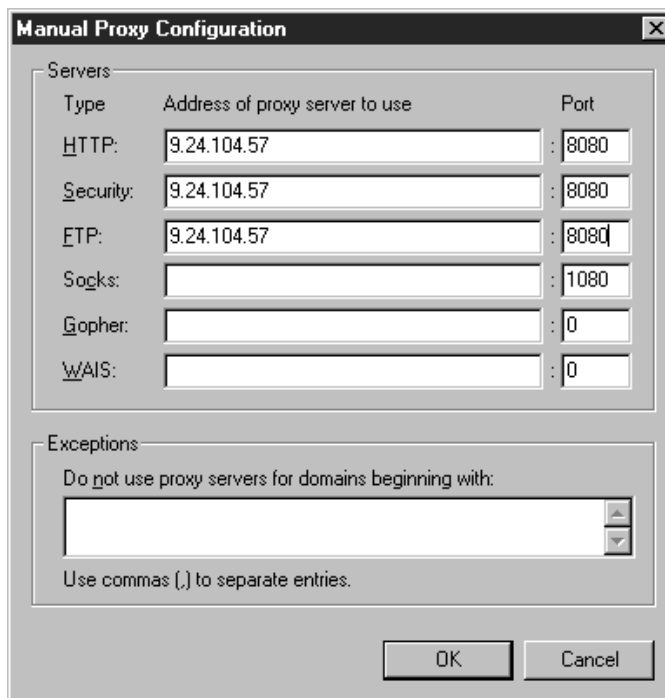


Figure 575. IBM Web Traffic Express Configuration in Netscape Navigator

Please see 13.6.3.1, "Browser Connection" on page 321 for details.

21.3.1.7 Further Proxy Configuration

In order to make the IBM Web Traffic Express more secure and improve its performance you should do some more configuration. The Web-based configuration allows you to change the following options:

Access Control	Set up access control for the server.
Administration of Users	Manage user ID and passwords.
Basic	Specify required settings.
Directories and Welcome Page	Set viewing options for the initial page, directory list contents, directory icons and user directories.
Error Message Customization	Customize error messages the server returns to clients.
Internal and External Java servlet settings	Define Java servlet settings for the server.
Languages and Encoding	Define file extensions for multi-format processing.
Logging and Reporting	Customize access log and error log and generate access log reports.
Meta Information	Name meta information files and directories.
PICS Services Configuration	Manage Platform for Internet Content Selection (PICS) rating services.
Request Processing	Specify how the server responds to an incoming request.
System Management	Specify parameters that will affect performance, work load, and SNMP.
Proxy Settings	Configure server as a proxy, specify caching settings, filters, and expiration dates.
Web Traffic Express Settings	Configure IBM Web Traffic Express server settings, such as cache refresh rate, PICS filter control, SOCKS servers and proxy performance.

Please refer to the *IBM Web Traffic Express for Multiplatforms User's Guide* for a full configuration of the IBM Web Traffic Express HTTP proxy server.

21.3.2 Using IBM Web Traffic Express on a Separate Host

We use the scenario shown in Figure 576 on page 545 for using IBM Web Traffic Express on a separate host behind the firewall.

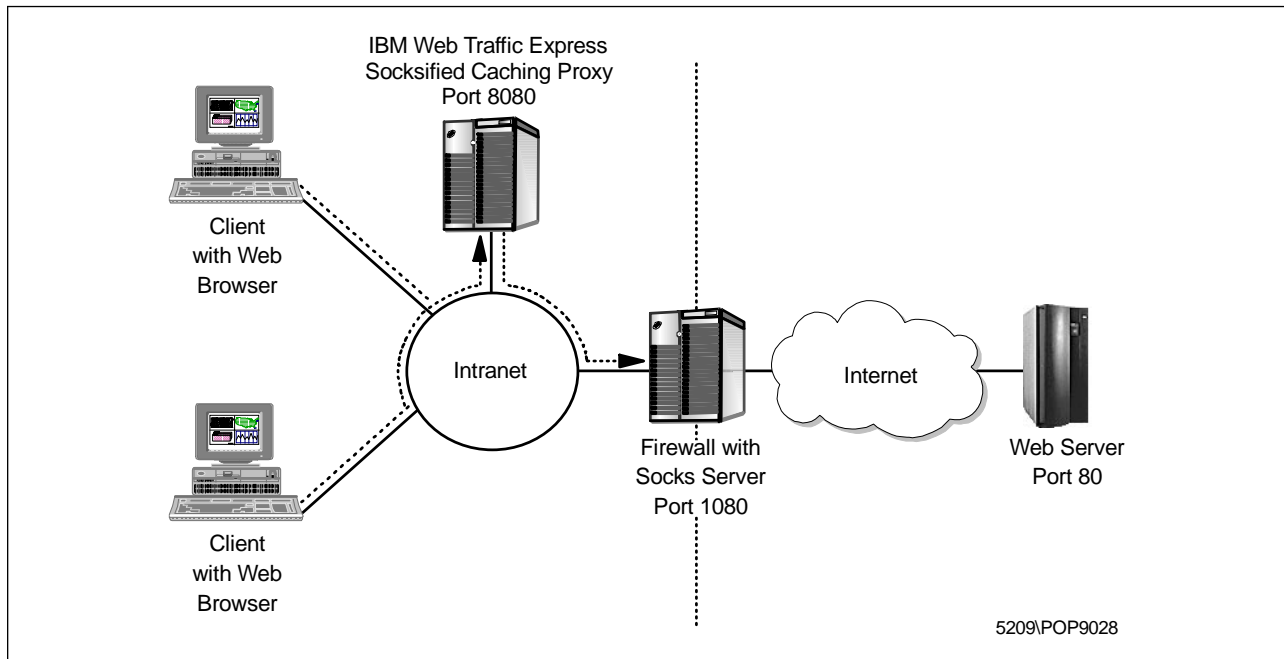


Figure 576. Scenario 2: IBM Web Traffic Express on a Separate Host

You have to do the following steps to install IBM Web Traffic Express on a separate host:

1. Disable the firewall HTTP proxy server (see 21.3.1.1, “Disable Firewall HTTP Proxy” on page 530).
2. Install IBM Web Traffic Express on a separate host. The installation procedure for the IBM Web Traffic Express on a separate host is the same as on the firewall (see 21.3.1.2, “Install IBM Web Traffic Express” on page 532).
3. Do some basic proxy configuration.
4. Define connection for IBM Web Traffic Express to access the firewall.
5. Point your Web browser to the HTTP proxy server (see 21.3.1.6, “Browser Configuration” on page 542, but use the IP address of the IBM Web Traffic Express host instead of the firewall IP address).
6. Further proxy configuration (see 21.3.1.7, “Further Proxy Configuration” on page 544).

21.3.2.1 Basic Configuration

You must configure the IBM Web Traffic Express HTTP proxy to use the firewall SOCKS server for all outgoing requests. To do so, start your Web browser and open the URL to your proxy server. In our case it's `http://9.24.104.70:8080/`. We have defined port 8080 as our proxy port. Your browser now shows the IBM Web Traffic Express main page (see Figure 567 on page 537). From this page select **Proxy Configuration**. You now will be asked for your user ID and password (see Figure 568 on page 537). Enter the appropriate data and select **OK**. You get the Proxy Configuration page shown in Figure 569 on page 538. Select **Proxy Server Settings** from this window. In the Proxy Server Settings window scroll down until you see the entry field for the SOCKS server.

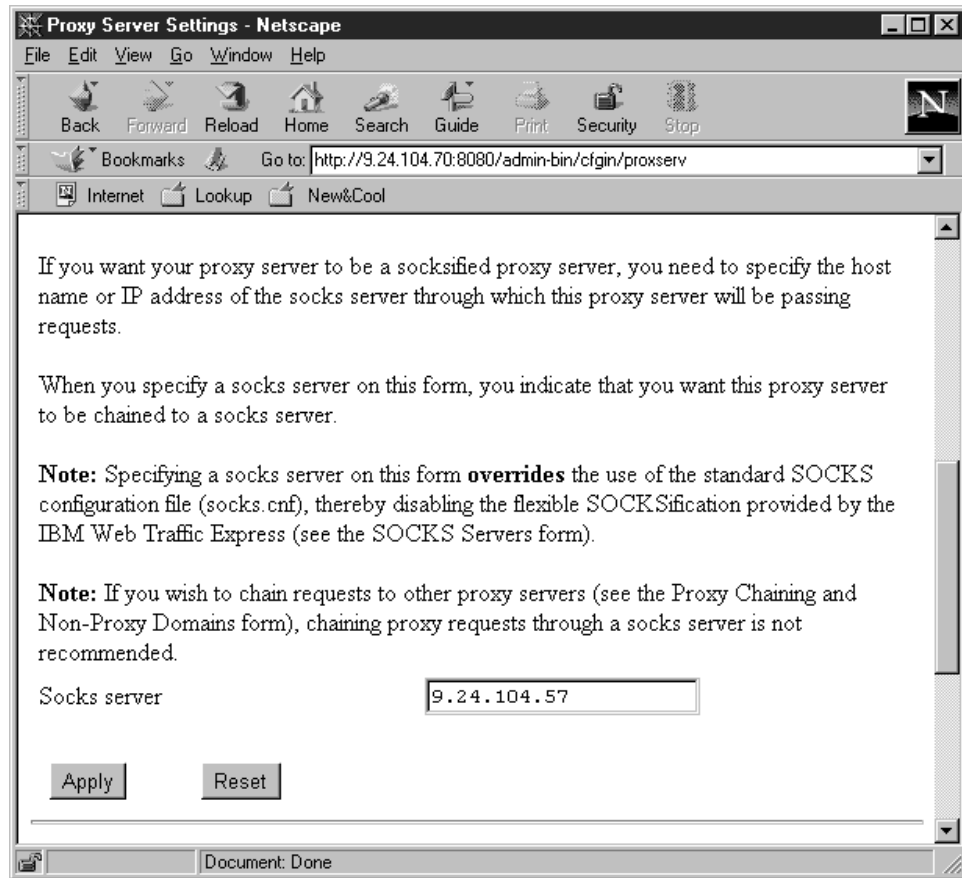


Figure 577. IBM Web Traffic Express Proxy Server Settings

Enter the IP address of your SOCKS server (which is on the firewall) and click on the **Apply** button. You will receive a settings confirmation page.

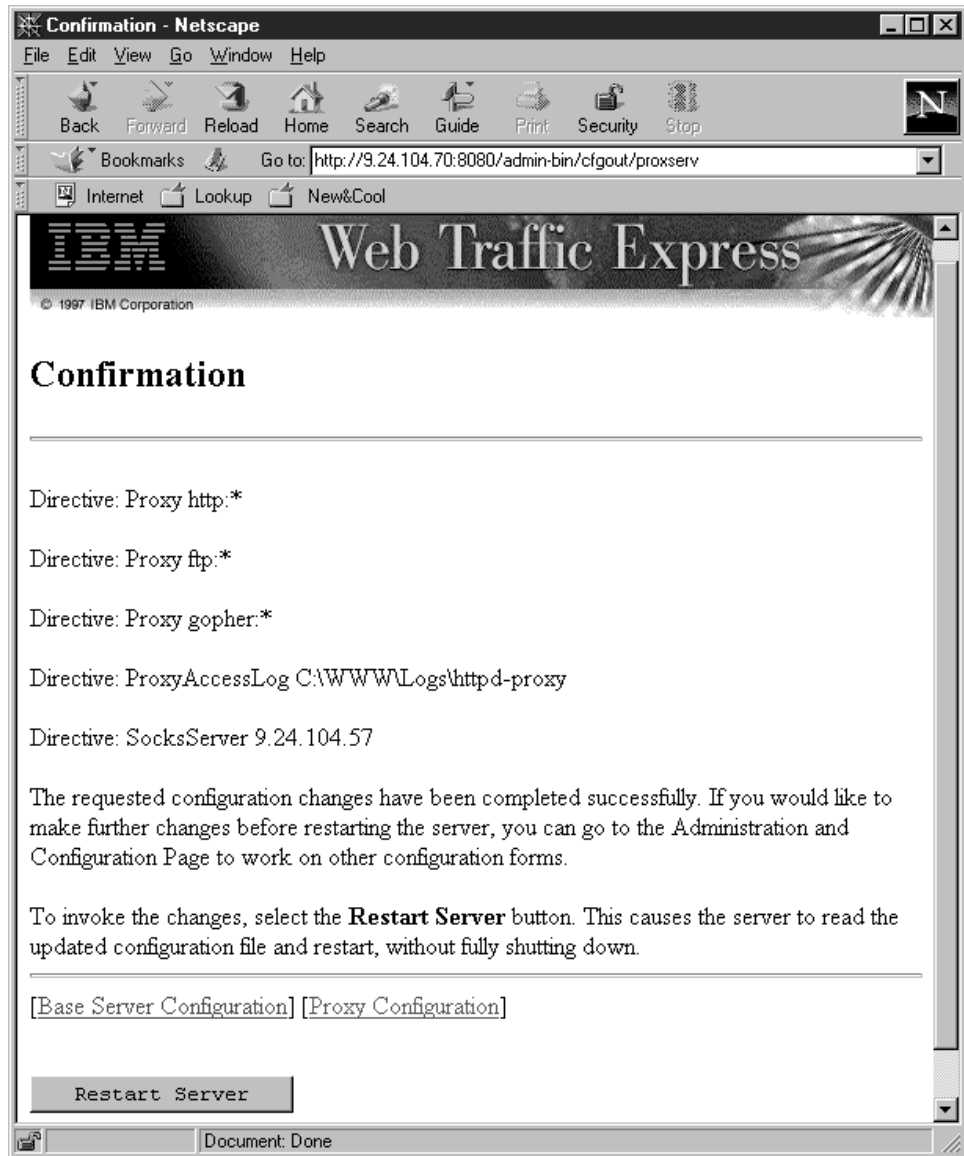


Figure 578. IBM Web Traffic Express Settings Confirmation Page

Review the new settings and click on the **Restart Server** button to finally activate your changes. You can now use the IBM Web Traffic Express for outbound connections.

Flexible SOCKSification

Every time a request is made using this method of SOCKS causes an interruption in the firewall while it routes the request to the correct destination.

However, flexible SOCKSification allows you to configure direct connections for specific IP addresses or hosts, and configure requests pass through the SOCKS server for other addresses. Now only requests that must go to the Internet must pass through the firewall, increasing efficiency of the network and reducing strain on the firewall or SOCKS server.

Please refer to the *IBM Web Traffic Express for Multiplatforms User's Guide* for a full configuration of the IBM Web Traffic Express HTTP proxy server.

21.3.2.2 Connection for Firewall Access

You have to specify two connections in order to allow your proxy server to connect to the firewall SOCKS server. This time you need to allow the proxy server, not your whole network, to use the SOCKS server (see 13.7.2.1, "Connections" on page 327).

First you create a connection from your proxy server to the IBM Firewall SOCKS server. Figure 579 shows the GUI window to build this connection.

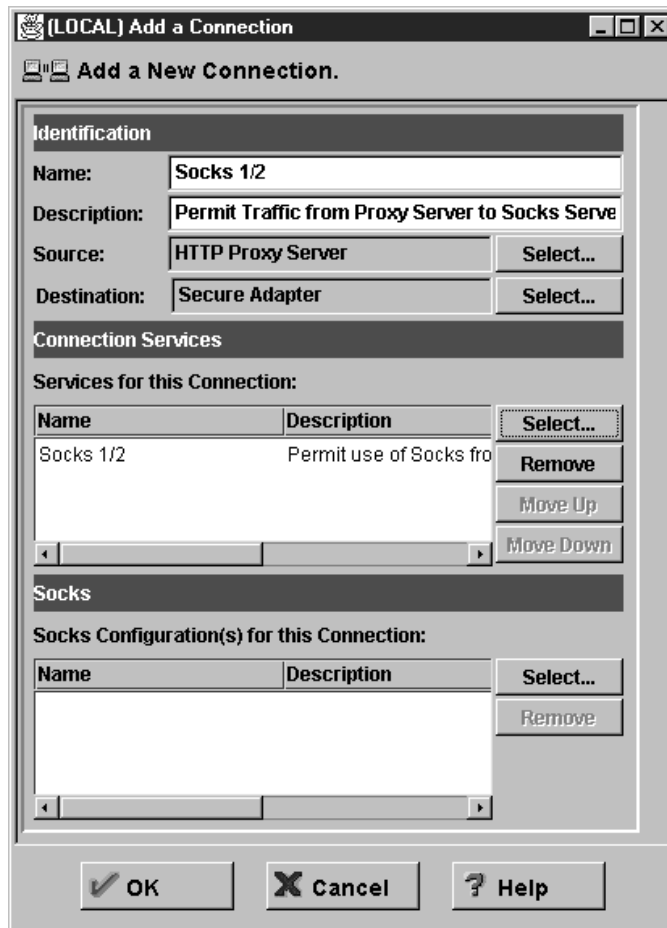


Figure 579. Connection from Proxy Server to SOCKS Server

As the next step you create a connection from the IBM Firewall SOCKS server to the non-secure network. Figure 580 on page 549 shows the GUI window to build this SOCKS outbound connection.

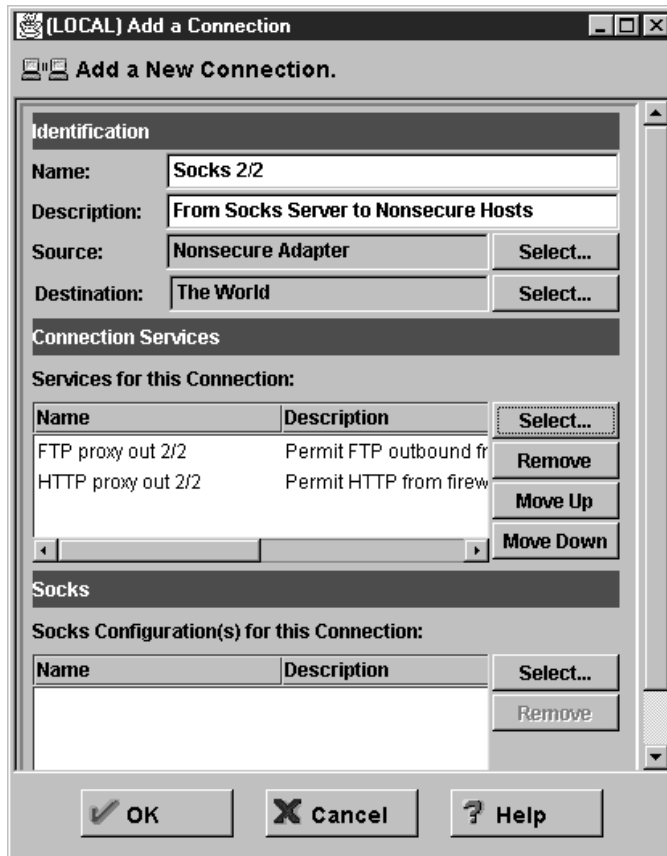


Figure 580. Connection from SOCKS Server to Non-secure Network

In this example, the permitted outbound services are FTP and HTTP.

As the last step you specify a SOCKS server configuration for connections between the proxy server and the non-secure network. Figure 581 on page 550 shows the GUI window for such a connection.

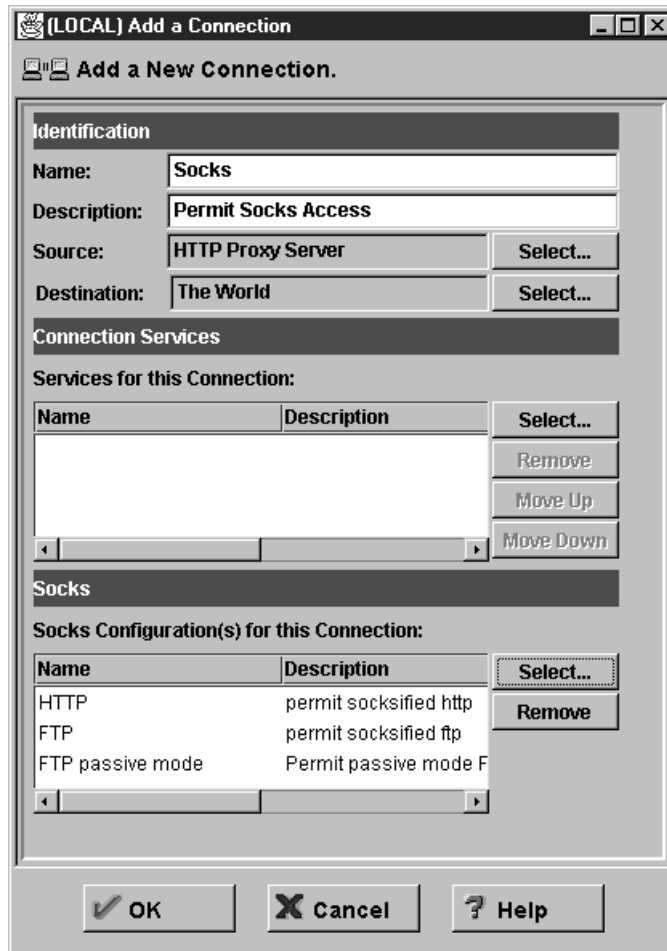


Figure 581. SOCKS Configuration from Proxy Server to Non-secure Network

Please see 13.7, “Using the SOCKS Server” on page 324 for a complete discussion of the SOCKS V5 server of the IBM Firewall.

21.4 MIMESweeper

Firewalls are essential for controlling access to our system, but to give a more secure environment, we need a tool that looks into the content of the data coming through onto the network. MIMESweeper from Integralis Technology is software that can help us to analyse the content of the data. It provides a solution for content security and it runs in Windows NT. There are two components of MIMESweeper: *WEBSweeper*, which deals a lot with Web content and *MAILsweeper*, which analyzes the content of e-mail. Since we consider MIMESweeper as an e-mail content security solution, then we will only discuss MAILsweeper. You can download the evaluation copy of MIMESweeper from <http://www.mimesweeper.com>.

Basically what MIMESweeper does is intercept the data (in the application layer), and process it through its standard procedures. At the end, MIMESweeper will determine whether to continue sending the data to the destination or send it to quarantine. The picture below, which is taken from the MIMESweeper Web site, describes the architecture of MIMESweeper.

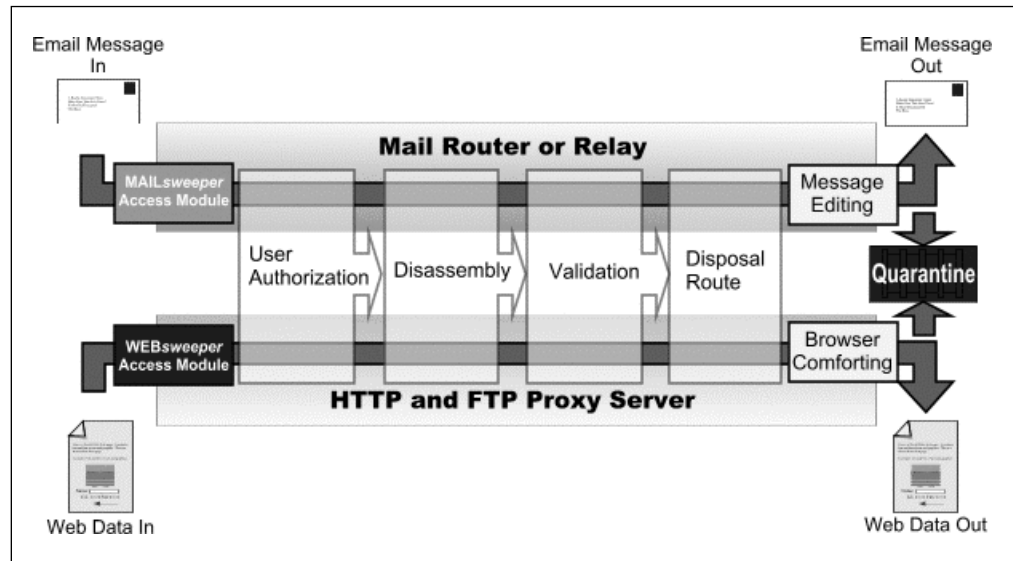


Figure 582. The Architecture of MIMESweeper

The list below provides brief information of all the processes. We only discuss the processes which are related with MAILsweeper.

- Email Message** This is the e-mail that will be accepted by MIMESweeper. MIMESweeper can process Lotus Notes, cc:Mail, SMTP and Novell Groupwise.
- Access Module** In the case of e-mail content, MIMESweeper will use the *mail access module* which is capable of understanding e-mail data.
- User Authorization** This process recognizes the source and the intended destination of a data item - *senders* and *recipients*. We can also create rules to perform certain action (for example, blocking the data) when a particular source or destination is detected in the data.
- Disassembly** When an item of data is received by MIMESweeper, it will be analyzed to see if it contains other data types. If it does, MIMESweeper will break down the data into its constituent parts. This is called *recursive disassembly*. For example, MIMESweeper will identify a ZIP file in the data and analyze its content.
- Validation** Validation is the process of checking the content of the e-mail data. One or more validators check each component of e-mail data for a certain content.
- Disposal Route** Once MIMESweeper has checked the item using its internal or plug-in validators, a *disposal route* is chosen. Possible routes include: delivery, quarantine, blocking, informing certain person, copying the data to certain person or editing the data.
- Email Message Out** Once the message has been validated, and assuming the message has not been blocked or quarantined, it is sent back into the e-mail system for normal delivery.

Quarantine

MIMESweeper supports up to 10 quarantine areas which can be used to hold suspect data. The data can be viewed by the MIMESweeper administrator.

Now that we know the architecture of MIMESweeper, we can begin with installation and configuration. We suggest the readers to read the manual of the product to get detailed information about how to install and configure MIMESweeper.

21.4.1 Installation

MIMESweeper is installed in the internal SMTP gateway. In our case, we install MIMESweeper in smtpsrv.itso.faire.com (9.24.104.243). See Figure 589 on page 556 for an example of mail configuration.

The installation of MIMESweeper is very straight forward. Run the installation application by running SETUP.EXE from the MIMESweeper CD-ROM or other media. The welcome screen will appear as shown in Figure 583.



Figure 583. MIMESweeper Installation Welcome Screen

Click on the **Next** button to continue the setup. The setup will check for an existing installation of MIMESweeper and then will ask for the modules that will be installed as shown in Figure 584 on page 553.

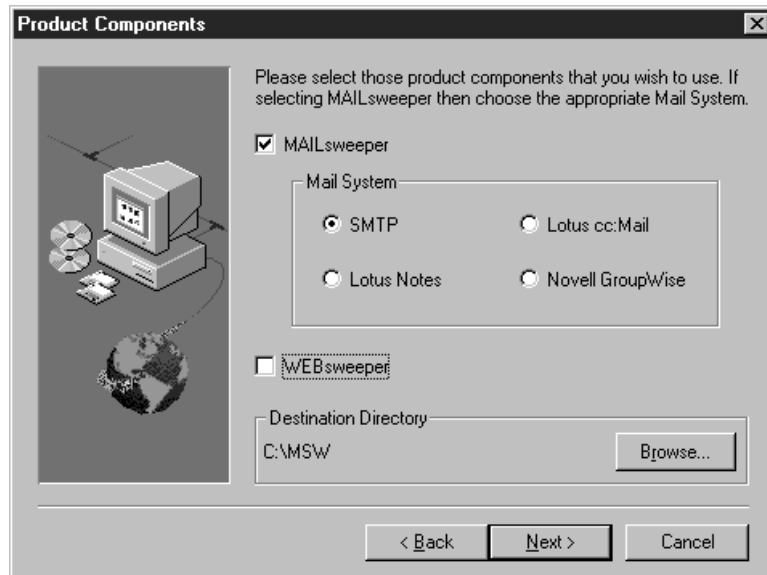


Figure 584. Product Component Screen

Since we use MIMESweeper only for e-mail content security, we choose to install MAILsweeper. Since Post.Office is used as a mail server (see Chapter 12, “Mail Handling” on page 269), we choose **SMTP** for Mail System. Click on the **Next** button after you have finished with all the options. Then, the setup will ask for the domain name as shown in Figure 585.



Figure 585. Post Office Details Screen

We have to enter the domain name of our internal network. This is the domain that our mail server is responsible for. In our case we enter `itso.faire.com`, which is our secure domain. After we click on the **Next** button, the setup program will ask for the account in our mail server. This account will be used by MIMESweeper to send information. This account will be used as a sender address. Figure 586 on page 554 shows the MAILsweeper account details screen.



Figure 586. MAILsweeper Account Details Screen

This account will be used by the disposal route process of MIMESweeper. Please make sure you create that account on your mail server. After we click on the **Next** button, we will get another screen which ask for the e-mail account to use as a destination address for information messages. The screen is shown below:



Figure 587. MAILsweeper Administrator Details

This account will also be used by the disposal route process. Usually the account postmaster is already created. The setup program will finish the installation by copying the programs to the destination directory. The last part of the installation is to enter the license key. We can do this through the process of installation or after the installation. The license screen is shown below.

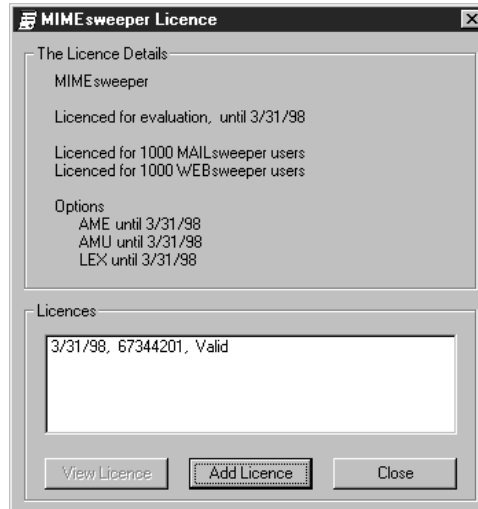


Figure 588. MIMESweeper License Screen

After we have finished with the installation, then we can configure the MIMESweeper. The scenario that will be discussed is the configuration of MIMESweeper on the SMTP gateway.

21.4.2 Configuration of MIMESweeper on the SMTP Gateway

The scenario that we will test is to block all junk mail. All mail from either *@junkmail.com or junkmail@* will be blocked. Before we change the configuration files of MIMESweeper, we will describe how to set up the environment of the SMTP gateway to work properly.

To refresh the configuration that we have in our lab, the diagram of our mail lab is shown in Figure 589 on page 556.

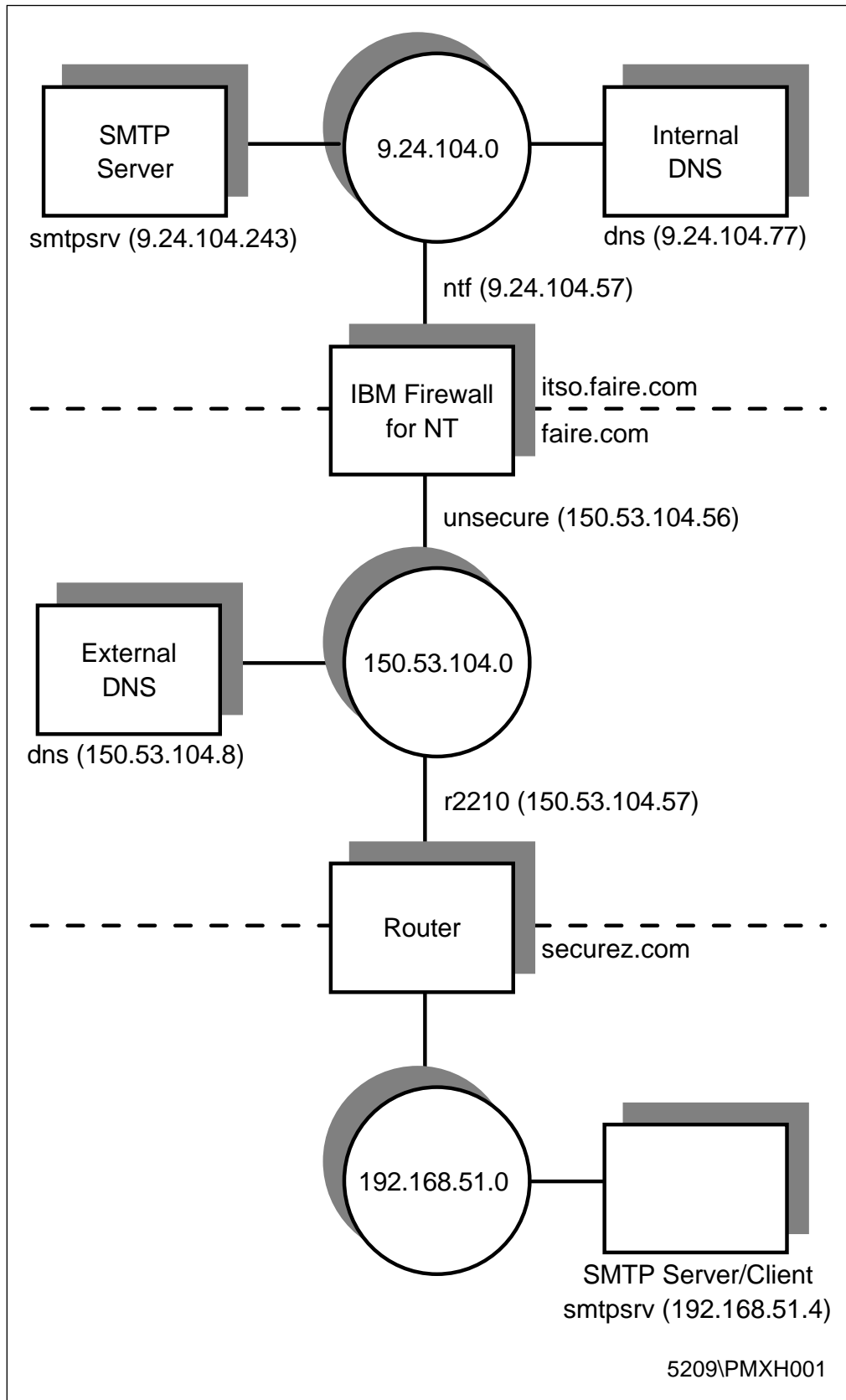


Figure 589. Example of Mail Configuration

The procedure to configure the MIMESweeper on the SMTP gateway is described below.

1. Pre-Installation.

Before we install the MIMESweeper on the SMTP gateway, there are a couple of things that we should check.

- Ensure that RPC services are running.
- If you want to use MIMESweeper to scan the viruses, then you should install and enable the anti-virus software.

2. Configure the SMTP gateway to listen to the new TCP port number.

In a normal mail configuration (without MIMESweeper), the use of the TCP ports looks like the following:

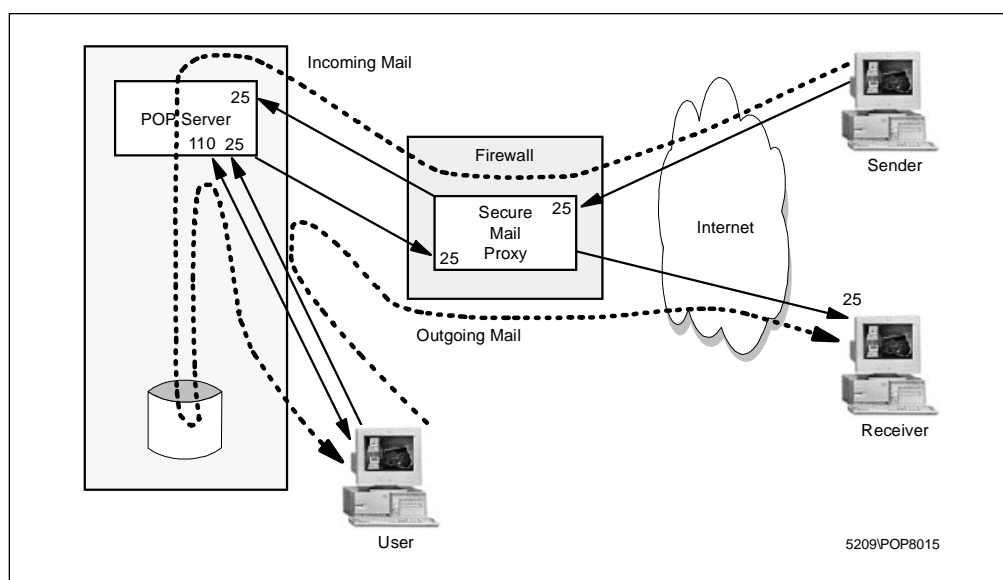


Figure 590. SOCKETS Value for the Post.Office

Since MAILsweeper uses port TCP/25 to receive the mail (remember, MIMESweeper will intercept our mail), we have to allocate a new port number for the internal SMTP gateway. In our case, we choose port 25000. The configuration of our internal mail server can be described in the diagram as shown in Figure 591 on page 558.

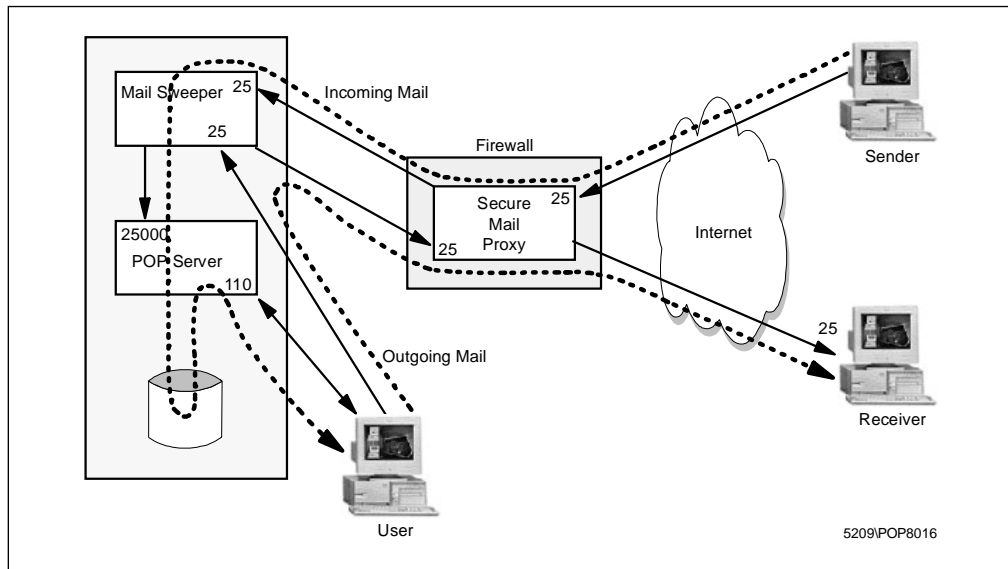


Figure 591. Configuration of Internal SMTP Gateway Using MIMESweeper

We configure the Post.Office to listen to the port TCP/25000 by changing the SOCKETS value in the NT registry which is located at:

HKEY_LOCAL_MACHINE\SOFTWARE\Software.com\Post.Office\SMTP-Accept\Config

Open the NT registry by running regedt32.exe. Go to the above location, double-click on the **SOCKETS** value to change the value and enter the new port number: 25000. The new value is shown in Figure 592.

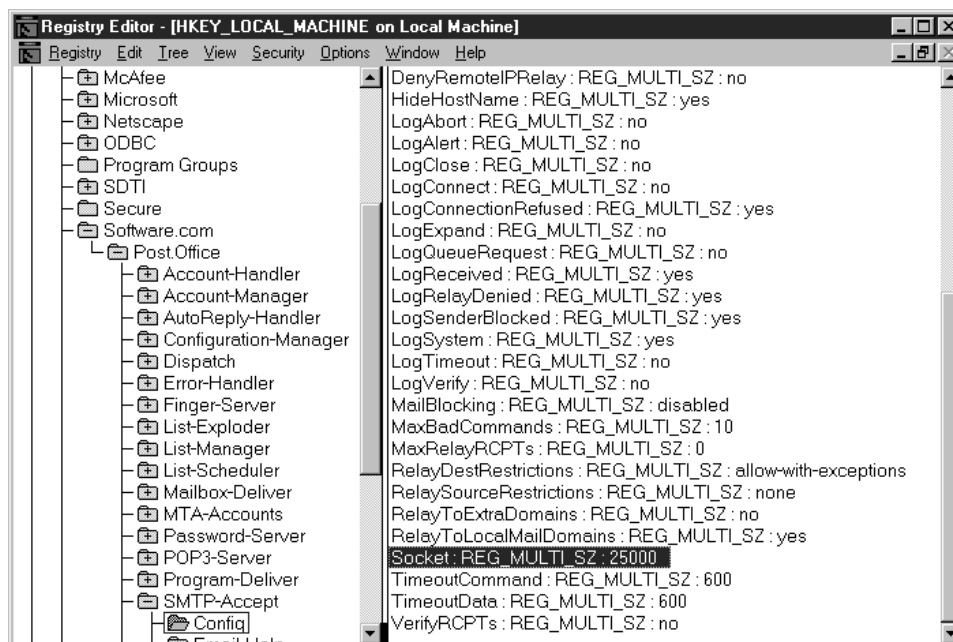


Figure 592. New SOCKETS Value for the Post.Office

Restart the Post.Office service and the mail server is ready.

3. Configure the gateway to forward all outgoing mail to localhost.

Now the incoming and outgoing mail will be handled by MIMESweeper but we want it to be transparent to our mail users. We have to change the route of the

outgoing mail. The outgoing mail will be sent to the localhost (127.0.0.1) and it will be taken care of by MIMESweeper since the MAILsweeper module Integralis SMTP Receiver, will listen to the port TCP/25. In the case of Post.Office, we change the route of the outgoing mail on the internal SMTP gateway by adding the following line:

```
*:[127.0.0.1]
```

It says that all outgoing mail will be sent to localhost port TCP/25. (refer to the user's guide of your mail server). Please note that MIMESweeper will work only with a mail server that has the capability to alter the port number.

4. Add MIMESweeper routing.

Since all mail is handled by MIMESweeper, then we must add new routings into the MIMESweeper configuration. The routing information will describe the way MIMESweeper routes all incoming and outgoing mails. For incoming mail, we set the route to send all mails to localhost port TCP/25000 (the new port number used by Post.Office). For outgoing mail, we set the route to send all other domains to the firewall.

To get the Integralis MAILsweeper dialog box, open **Control Panel** and double-click on **MAILsweeper** icon. Click on **Routing** to get the routing dialog box.

In our case, we add our internal domains, faire.com and itsso.faire.com, to the MIMESweeper configuration and send all other domains to the firewall at 9.24.104.57. We have to add faire.com to the routing information because the Secure Mail Proxy will not rewrite the destination address of the incoming mail. That will be internal SMTP gateway's responsibility. The new configuration is shown in Figure 593.

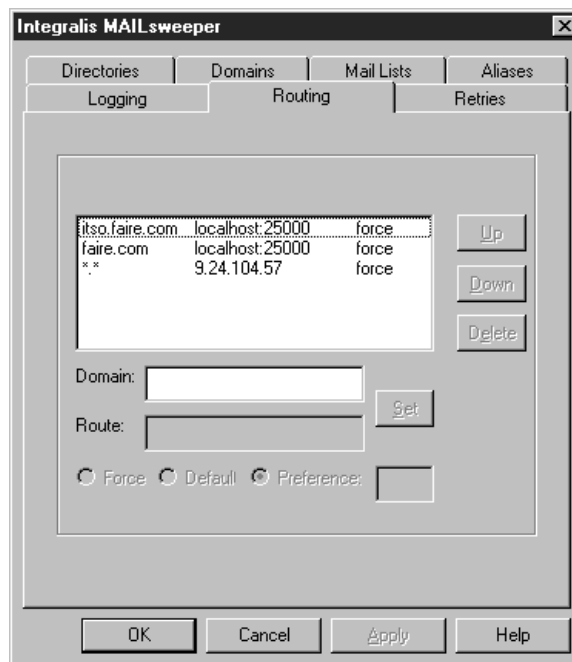


Figure 593. New Routing Information in MIMESweeper Configuration

5. Change the MIMESweeper configuration files. There are three main files in the directory C:\MSW\Config that we have to change to enable the MIMESweeper blocking the junk mail. These files are described below.

a. Validate.cfg

The purpose of this file is to provide MIMESweeper with the information it needs to perform validation of the raw data found in a mail message as a result of recursive disassembly. We uncomment the following lines to enable MIMESweeper to use the built-in validator, AMUcheck.

```
[AMU]
AuthFile=C:\MSW\Config\AUTHFILE.TXT
```

The built-in validator, AMUcheck, performs validation by comparing the source and destination addresses of a message against a set of *access* rules. These rules specify different address combination that may match with the addresses of the message. Addresses are specified using the generic address format *user@location*. The lines above tell the validator to look at the AUTHFILE.TXT that is described next.

b. AUTHFILE.TXT

The AMUcheck uses the file AUTHFILE.TXT as specified above. Part of the files are shown below:

```
;
; Example AMUcheck Authorisation file for MIMESweeper SMTP.
;
RESPONSE allow
RESPONSE allow_in
RESPONSE allow_out
RESPONSE copyadministrator PRIORITY 2
RESPONSE isjunkmail PRIORITY 3
RESPONSE deny PRIORITY 4

;
; A default rule which allows everything.
;
FROM *@* TO *@* allow

;
; Spot messages that are entering the company.
;
FROM *@*
    TO *@faire.com allow_in
    TO *@itso.faire.com allow_in
;
; Spot messages leaving the company.
;
FROM *@itso.faire.com To *@* allow_out

;
; Everything from annoying company is junk mail.
;
FROM *@junkmail.com TO *@* isjunkmail
FROM junkmail@* TO *@* isjunkmail
```

The file tells that all mail from either **@junkmail.com* or *junkmail@** destined to **@** will be recognized as junk mail. The priority of the junk mail is 3. The disposal action that will be taken according to the response in the above file is described in the next file.

c. Mimeswp.cfg

The response that we use if the MIMESweeper receives junk mail is isjunkmail. To describe the configuration of that response, we add new lines in the main configuration file of MIMESweeper as shown below:

```
;
; Disposal Section.
;
...
[Disposal]
ISJUNKMAIL=DeleteJunk
...
;
; Disposal Actions Sections.
;
...
[DeleteJunk]
Quarantine=JunkQuarantine
Inform=JunkList
...
;
; Inform sections.
;
...
[JunkList]
FromAdr=%SERVER%
ToAdr=%ADMIN%
Subject=The mail is a junk mail.
Body=C:\MSW\Config\junkmail.TXT
WithCopy=TRUE
...
;
; Quarantine Control Sections.
;
...
[JunkQuarantine]
Comment=Blocked junk message from %SENDER%
Area=JunkQuarantine Messages
...
;
; Quarantine Area Sections.
;
...
[JunkQuarantine Messages]
Location=C:\MSW\Qtine\Encrypted\
File=C:\MSW\Qtine\Encrypted\QUARNTNE.LST
...
```

The main section is Disposal, which describes the disposal action of every response. In our case we specify that ISJUNKMAIL will use the DeleteJunk disposal action. Then in the Disposal Actions Section we describe the DeleteJunk. It tells that we will send the mail to quarantine and inform the *postmaster* about the junk mail.

In the Inform sections, we describe the sender address and destination address for the information mail. MIMESweeper provides some variables to use such as %SERVER% and %ADMIN% for the mailsweeper and postmaster account respectively. We add the subject and the body of the

mail is put in another file called junkmail.txt. The contents of junkmail.txt is shown below:

Mailsweeper found a junk mail.
Please take necessary actions.

The next sections describe the area of the quarantine. We call the quarantine of junk mail as JunkQuarantine and the area as JunkQuarantine Messages.

The configuration of MIMESweeper files is the last step of our setting. Now, we are ready to test the MIMESweeper configuration.

21.4.3 Test the Configuration

To monitor incoming mail to the MIMESweeper, we have to use the MIMESweeper manager. We can open it through the START menu and choose **MIMESweeper Manager**. We test the environment by sending mail from junkmail@a.com and a@junkmail.com. MIMESweeper captures the mail and shows the disposal actions that have been taken, as shown in Figure 594.

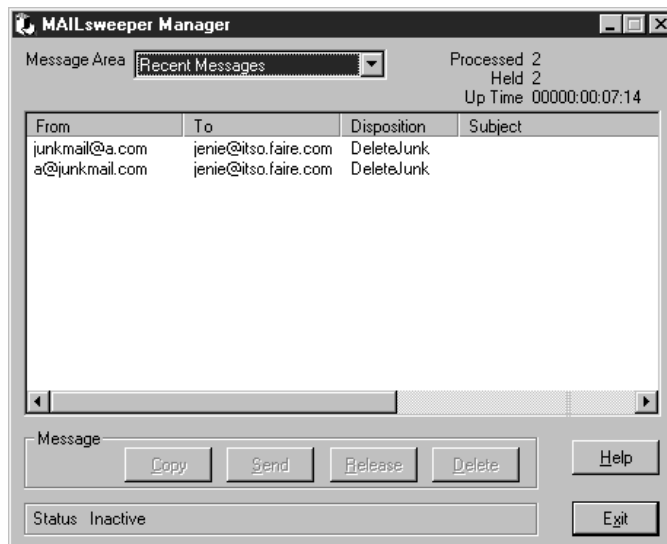


Figure 594. Recent Messages Screen of MIMESweeper

MIMESweeper used the DeleteJunk disposal action for these two junk mails. From the definition in the Mimeswp.cfg file, we know that the mail has been quarantined. We can see the mail in our quarantine area by changing the message area to JunkQuarantine Messages.

We can copy, delete, send or release the mail from the quarantine area. The last action of the disposal configuration is to send the information mail to the postmaster. We use the Netscape mail client to view the mail as shown in Figure 595 on page 563.

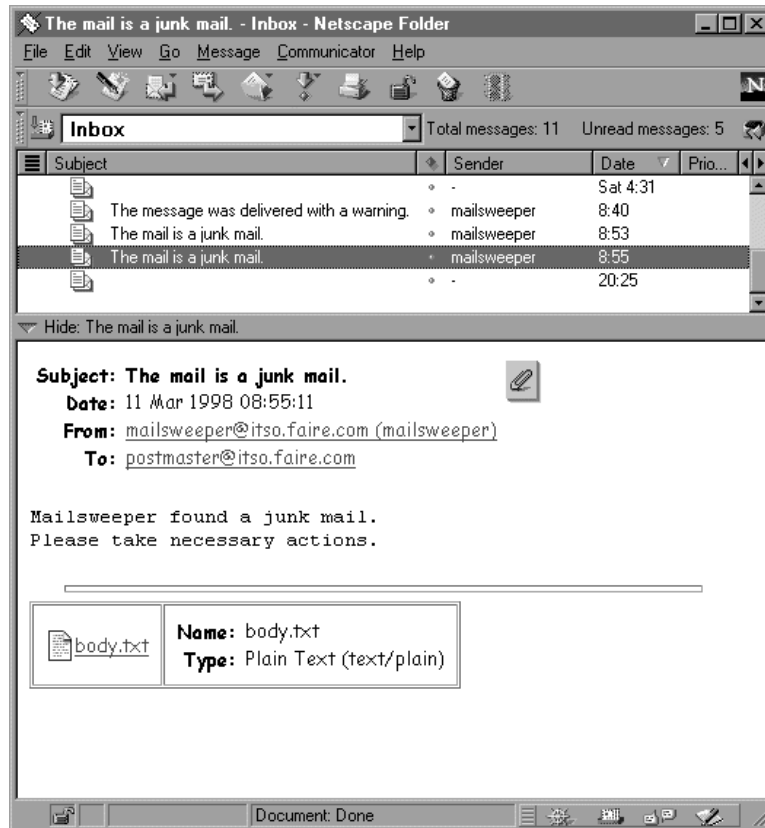


Figure 595. Information Message from MAILsweeper

21.5 SecurID Authentication

In this section we show an example of how to implement the SecurID protection to your network.

The ACE/Server from Security Dynamics (<http://www.securitydynamics.com>) implements SecurID and/or SoftID identification and authentication of users on TCP/IP networks.

It uses SecurID tokens, which are small handheld microprocessed devices and/or SoftID, which is software running in the user's PC that displays codes changing in a specific interval of time, generally 60 seconds.



Figure 596. SecurID Tokens

21.5.1 How the ACE/Server Implements Strong Authentication

The ACE/Server uses a superior scheme of authentication because it requires two factors instead of just one. To gain access to the protected system in your network, users should enter a SecurID PASSCODE, which is made up of:

- A secret PIN number.
- The code displaying on the token assigned to the user. This code is valid just for a period of time. Generally 60 seconds.

If a hacker is able to catch this PASSCODE when it is transmitted over the Internet or the intranet, it will not do him/her any good since it will change within a few seconds. If the SecurID card is lost, it cannot be used without knowing the PIN. There is no need to enforce convoluted passwords rules to the users. With these two types of authenticity, you do really improve the security of your network environment.

We don't need to install any additional software on the firewall machine because the IBM Firewall has the ACE/Agent code built in. In the IBM Firewall we can use the SecurID and/or SoftID authentication for:

- Secure and non-secure Telnet
- Secure and non-secure FTP
- Secure and non-secure SOCKS
- Secure HTTP
- Secure and non-secure Firewall Administration

It is possible to have the ACE/Server running on the Firewall machine, but for security issues our recommendation is to install the ACE/Server on a separate machine in the secure network.

21.5.2 Configuring the ACE/Server

We received a CD-ROM with the ACE/Server V.3.1 and ACE/Agent codes, a diskette with the ACE/Server license code, a diskette with the tokens record, and SecurID cards from Security Dynamics. To install the ACE/Server:

1. Log on as a Windows NT administrator.
2. Insert the ACE/Server V3.1 CD-ROM into the CD drive.
3. Insert the diskette labeled "ACE/Server V3.1 Master Server License" into the diskette drive.
4. On the Windows NT Start menu click on **Run**.
5. In Run dialog enter `d:\aceserv\nt_i386\setup.exe` and click on **OK**. Follow the instructions on the screen. See the *ACE/Server V3.1 for Windows NT Installation Guide* for more information about the ACE/Server V3.1 installation.

Now we will explain all the configuration steps necessary on the ACE/Server to permit SecurID authentication for IBM Firewall users. We can divide the configuration process into six steps as follows:

1. Create clients.
2. Create groups and activate them on the client machine.

3. Set system PIN parameters.
4. Create users, assign tokens, and add users to groups.
5. Start the ACE/Server services via the Windows NT Control Panel.
6. In the firewall machine, create the connection between the secure interface of the firewall and the ACE/Server.

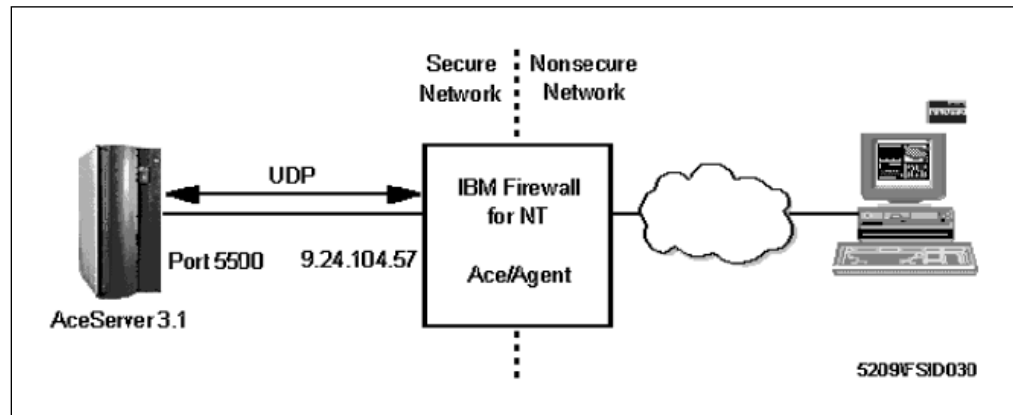


Figure 597. Configuration

21.5.2.1 Creating Clients

Here we describe how to create a client (in our case, the IBM Firewall) that will communicate with the ACE/Server.

1. From the Windows NT desktop, select **Start** → **Programs** → **ACE Server** → **Database Administration-Host Mode**.
2. Select **Client** → **Add Client**.

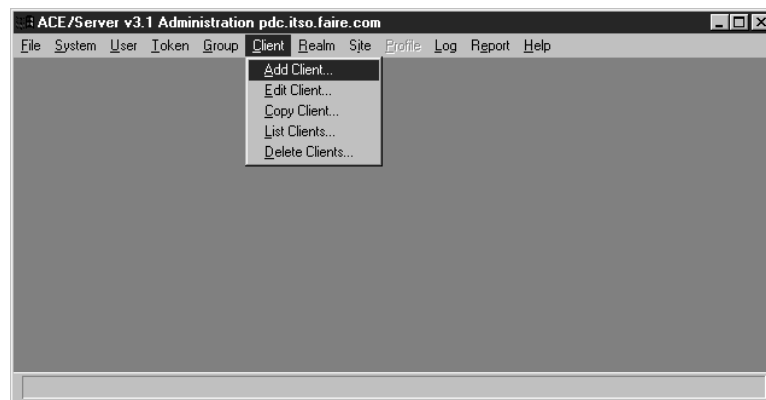


Figure 598. ACE/Server Administration Panel

3. Enter the name of the firewall machine. When you press Tab to exit the Name field, the IP address of the Firewall machine will automatically be displayed in the Network address field, based on the information you have in your DNS server or local hosts file.
4. Select the client type. As the IBM Firewall supports just the standard SecurID authentication (Enter PASSCODE, PASSCODE Accepted and Access Denied), you have to select **Single Transaction Comm Server**.

5. Select the **SDI** encryption type, which is the encryption type used by the IBM Firewall ACE/Agent to communicate with the ACE/Server.
6. Click on **OK**.

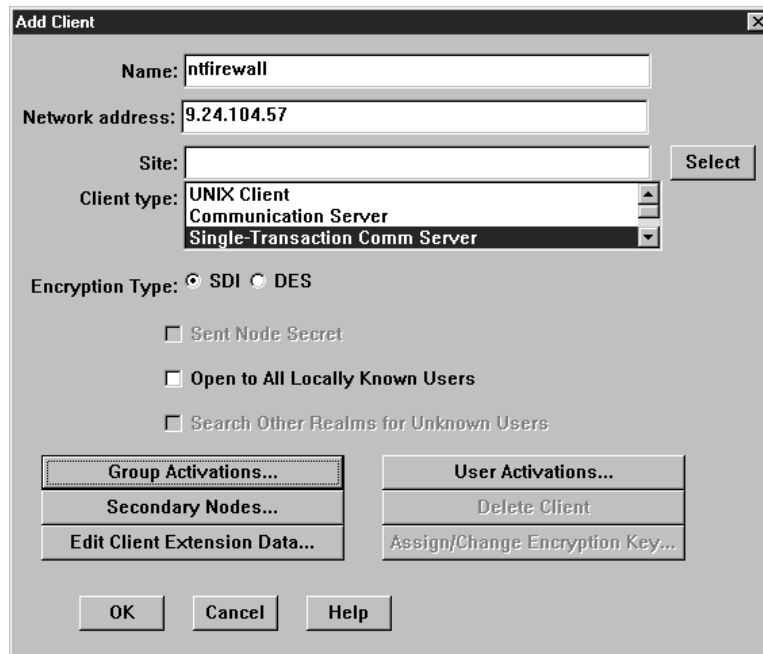


Figure 599. Add Client Dialog

21.5.2.2 Creating Groups

Now that you have created the client, you should create a group that will have the users authenticated by the ACE/Server.

1. From the Administration panel, select **Group** → **Add Group**.
2. Enter the name of the group you want to activate at the client. In our case we created the group Firewall Users.

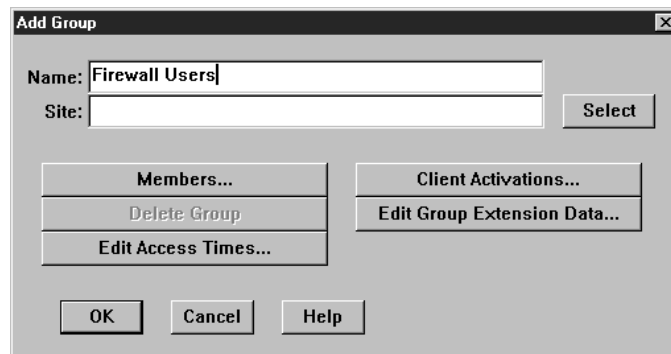


Figure 600. Add Group Dialog

3. Click on **Client Activations** and on **Yes** at the confirmation dialog.
4. Select the firewall machine and click on **Add Client**.

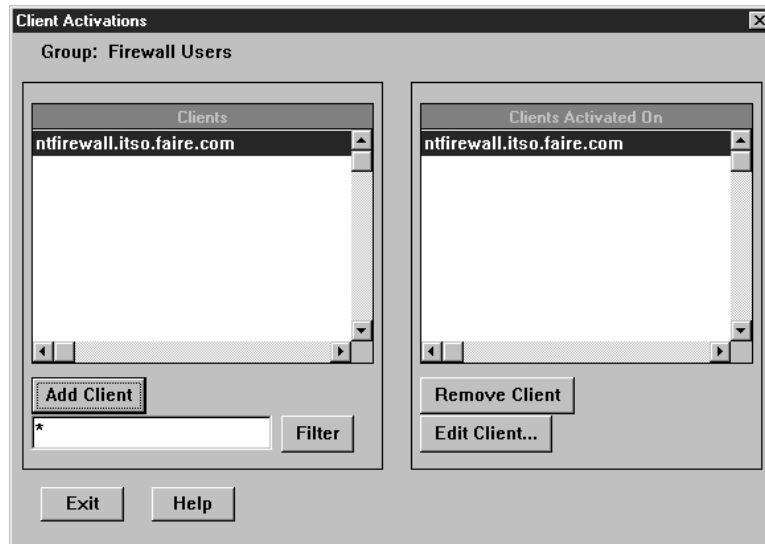


Figure 601. Client Activations Dialog

5. Click on **Exit**.
6. At the Add Group dialog, click on **OK**.

21.5.2.3 Setting PIN Parameters

Now you should set up the PIN parameters you want to be used by the firewall users.

From the Administration panel, select **System** → **Edit System Parameters**.

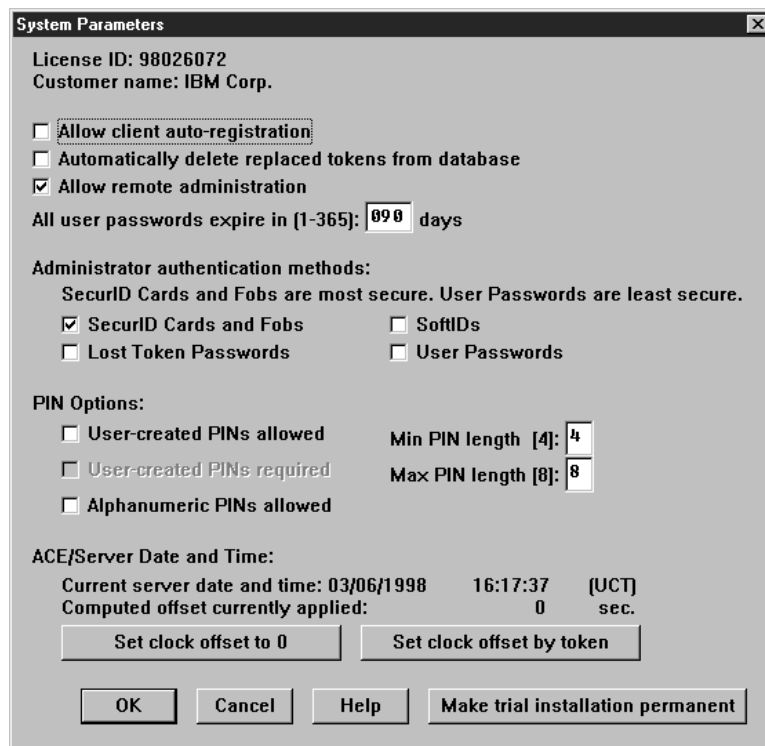


Figure 602. System Parameters dialog

In this panel we just deselected the **User-created PINs allowed** because the ACE/Agent in the IBM Firewall does not support the user creating his/her PIN number. For the other values we used the defaults. Click on **OK**.

Note: Fobs are the SecurID Key Fob tokens (key-chain type) as shown in Figure 596 on page 563.

21.5.2.4 Creating Users

Following are the steps to create the firewall users, how to assign tokens for them and how to put the users in the Firewall Users group.

1. From the Administration panel, select **Users** → **Add User**.
2. Enter the first and last name of the user and the default login of this user. The **Local User** option is selected by default. The **Remote user** option is selected when we are adding a user that is already configured in an another ACE/Server. In our case we configured just local users.

Serial Number	Type	Status
---------------	------	--------

Figure 603. Add User Dialog

3. Click on **Assign Token**.
4. Click on **Yes** at the confirmation dialog.
5. The Select Token dialog will show the serial number of the first unassigned token from the tokens record (shipped with the ACE/Server product). See the *ACE/Server Installation Manual for Windows NT* for more information about the tokens record.

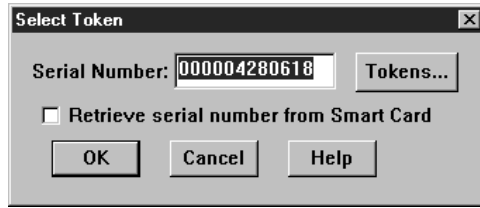


Figure 604. Select Token Dialog

6. Click on **OK**.
7. From the Add User dialog, Figure 603 on page 568, select the token that has just been assigned and click on **Edit Assigned Token**. You will be presented with the panel of Figure 605.



Figure 605. Edit Token Dialog

8. Click on **Resynchronize Token**.
Synchronization here means to adjust the clock in the token with the clock in the ACE/Server in order to ensure that the pseudorandom code displayed by a user's token is the same code generated by the ACE/Server software.
9. Enter the code displayed on the token and click on **OK**.
10. Wait for the code to change and enter the new code.

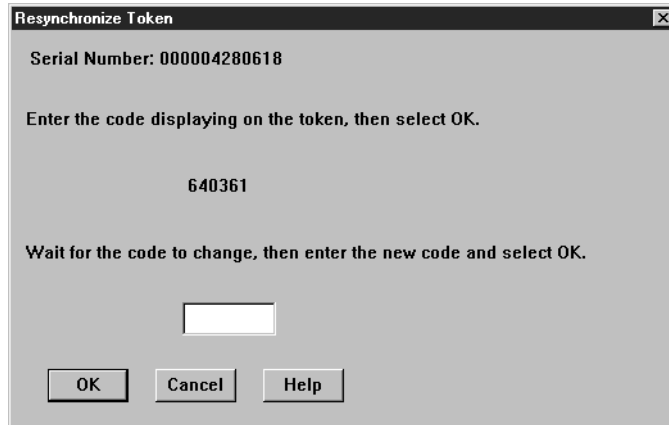


Figure 606. Resynchronize Token Dialog

11. Click on **OK**.
12. From the Edit Token dialog, Figure 605 on page 569, click on **Set PIN to Next Tokencode**.



Figure 607. Set PIN to Next Tokencode Dialog

13. Enter the code displaying on the token and click on **OK**. Your PIN number will be the first 4 digits of the next tokencode: the one that will appear after the actual 60 seconds period expires.

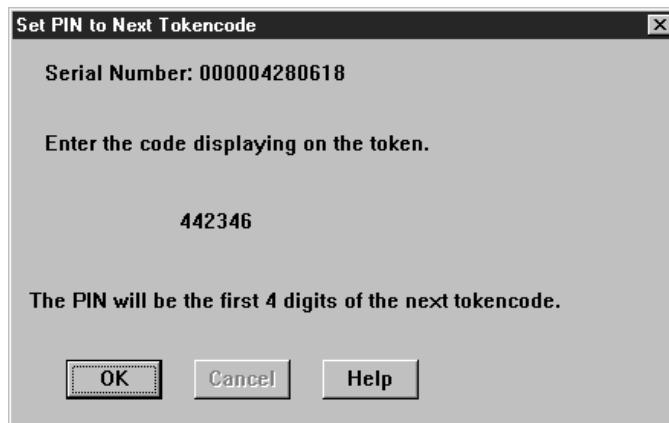


Figure 608. Set PIN to Next Tokencode Dialog

14. After writing down the PIN number in a secure place, click on **OK**. You'll have to tell this PIN number to the user when you give him/her the token.
15. From the Edit Token dialog, Figure 605 on page 569, click on **OK**.
16. From the Add User dialog, Figure 603 on page 568, click on **Group Memberships**.
17. Select the Firewall Users group and click on **Join Group**.

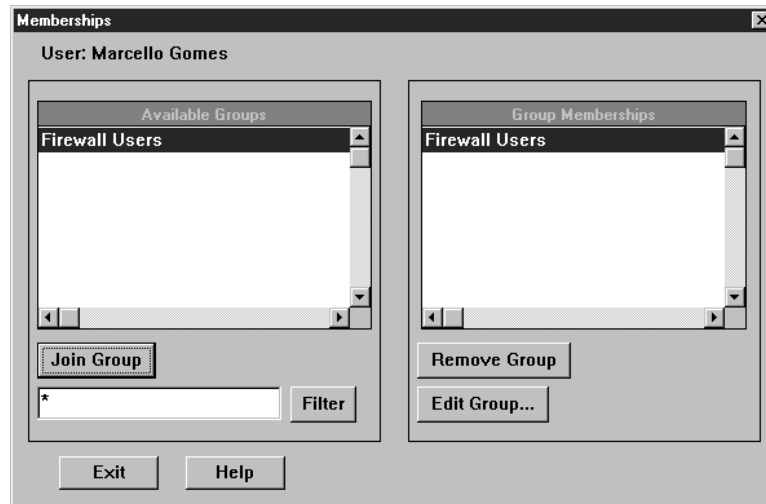


Figure 609. Memberships Dialog

18. Click on **Exit**.
19. From the Add User dialog, click on **OK**.

21.5.2.5 Starting the ACE/Server Services

Following we show the steps to start the ACE/Server services.

1. From the Windows NT desktop, select **Start** → **Programs** → **ACE Server** → **Configuration Management**.

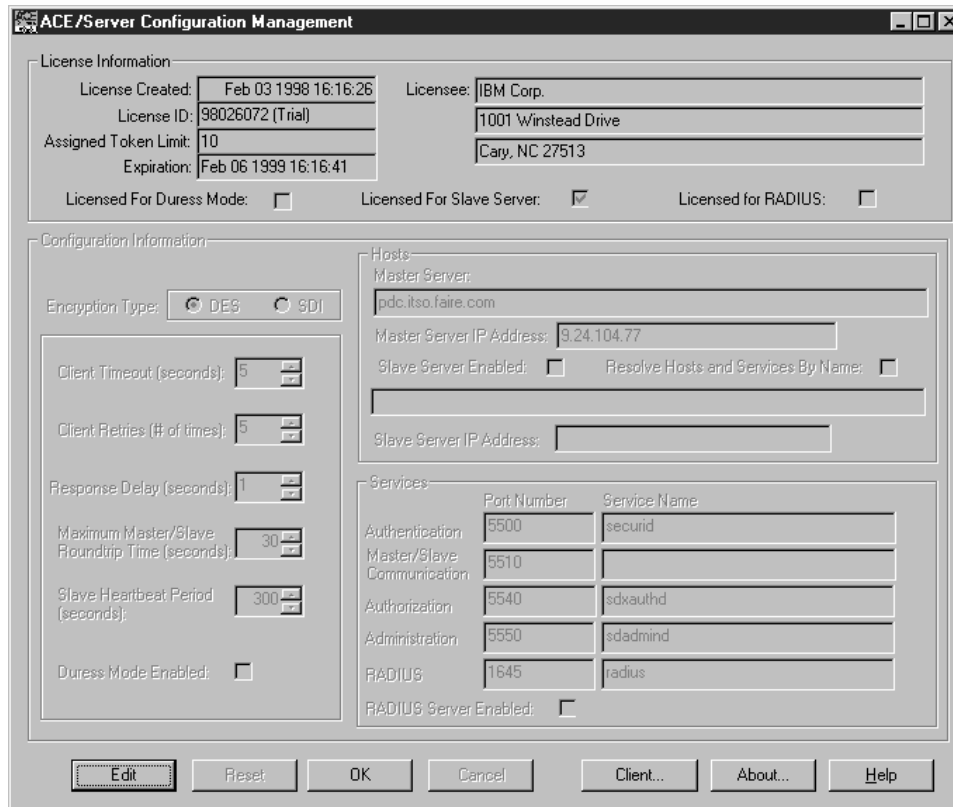


Figure 610. ACE/Server Configuration Management Dialog

2. Click on **Edit** and click on **OK** at the reminder dialog.
3. Change the encryption type to SDI which is the type of encryption that the ACE/Server will use to communicate with the IBM Firewall.

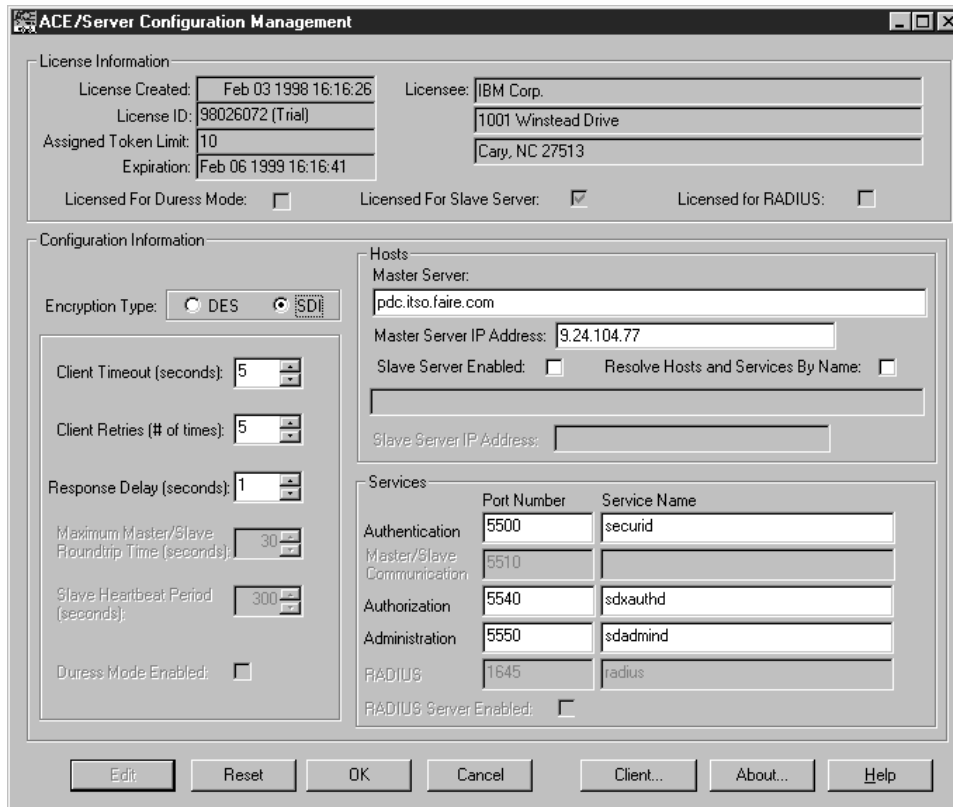


Figure 611. ACE/Server Configuration Management Dialog

4. Click on **OK**.
5. Click on **OK** at the warning dialog.
6. Save the changes.
7. From the Windows NT Control Panel window, click on the **ACE/Server** icon.

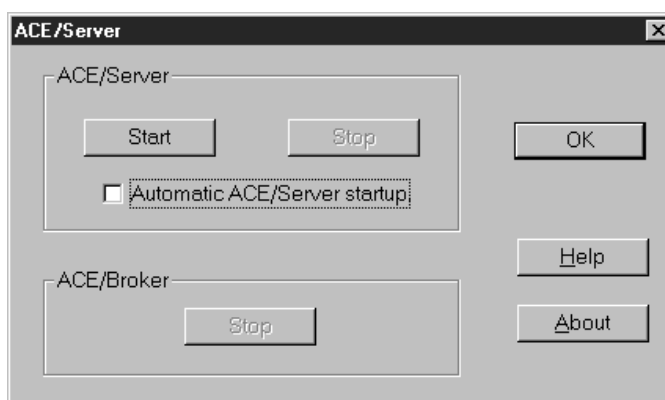


Figure 612. ACE/Server Dialog

8. Click on **Start**.
9. Wait for the message ACE/Server started and click on **OK**.
10. Copy the ACE/Server configuration file `sdconf.rec` from the `\ACE\data` directory on the ACE/Server machine to the `\WINNT\System32\` directory on the IBM

Firewall machine to allow the IBM Firewall to communicate with the ACE/Server.

21.5.2.6 SecurID Connection

In order to allow the IBM Firewall to use the ACE/Server services, you should configure a connection between the secure adapter and the ACE/Server.

Note: For security reasons, the ACE/Server should be in the secure network.

1. Create a network object for the ACE/Server. See Chapter 8, "IBM Firewall Rule Base" on page 143 for more information about creating network objects.
2. Create a connection where the source object is the secure adapter and the destination object is the ACE/Server.
3. Select the predefined service **SDI authentication** for this connection. This service uses the UDP protocol on port 5500 for firewall queries and ACE/Server replies.
4. Go to the Connection Activation panel, select **Regenerate Connection Rules** and click on **Execute**.

After all these steps, any user who is configured in the firewall machine to use SecurID authentication (see Figure 613 on page 575) will be able to take advantage of this strong authentication mechanism.

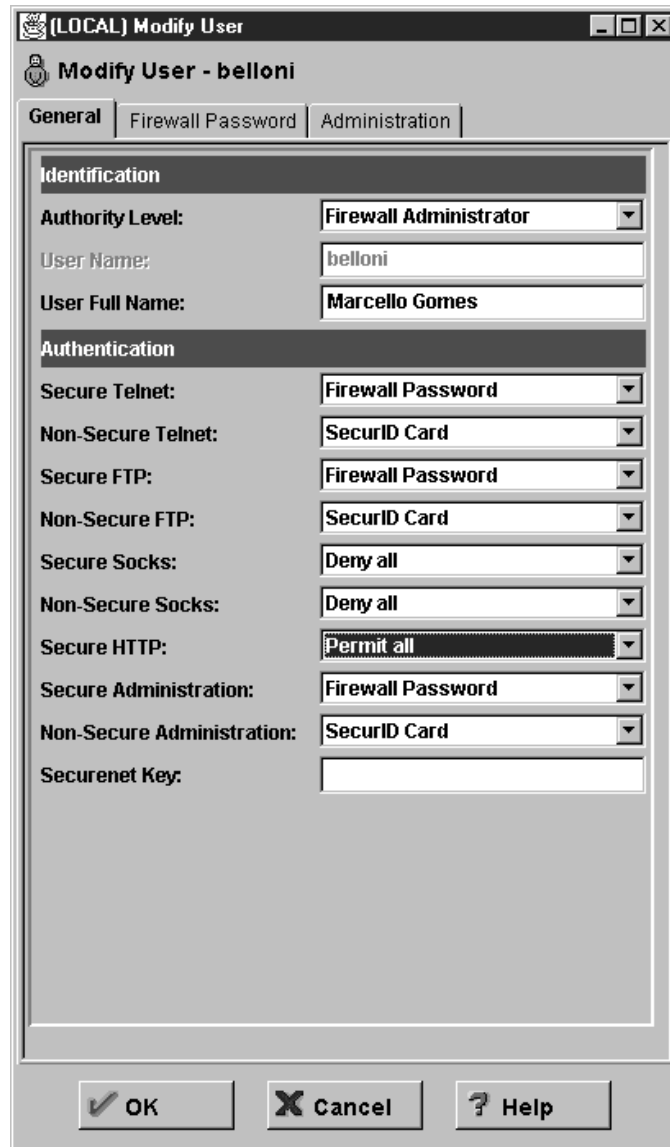


Figure 613. Configuring User For SecurID Utilization

The utilization of the SecurID token is very easy. When prompted for a PASSCODE as in Figure 614 on page 576, enter your PIN number followed by a comma, and next, enter the code displayed in your SecurID token.



Figure 614. SecurID Authentication

21.6 TME 10

In the world of network computing, corporations need an open, scalable system management tool to ensure that their IT infrastructure is running well to support their businesses. Tivoli Management Environment (TME) from Tivoli, a subsidiary of IBM Corp., provides an enterprise-wide, yet easy-to-use system management tool to address that issue. TME 10 was built with the concept of *Framework*. The framework provides an interface between management module and managed resources such as databases, applications, network and systems. The framework was written by pure object-oriented language which is compliant with Common Object Request Broker Architecture (CORBA), the common object-oriented architecture developed by Object Management Group (OMG).

In the case of the firewall, we will use TME 10 to manage the firewall machine. Using Tivoli, we can proactively resolve the problem that the firewall might have during its operation. We will use the Tivoli console to monitor events coming from the firewall and take necessary actions to resolve the problems.

Tivoli has many modules which can manage a broad base of commercial applications, such as SAP, Lotus Notes, and Netscape but not IBM eNetwork Firewall for Windows NT. Therefore, we can't use the modules from Tivoli to manage the IBM Firewall. We will use the capability of TME 10 Enterprise Console, one of the Tivoli core applications to monitor the IBM Firewall. And instead of using the full-blown framework, we will use Lightweight Client Framework (LCF) to execute commands into the firewall. It means that we will use the *non-secure delivery* event, which uses the standard Interprocess Communications (IPC) process. Figure 615 on page 577 shows the diagram of our configuration. The explanation of each component follows.

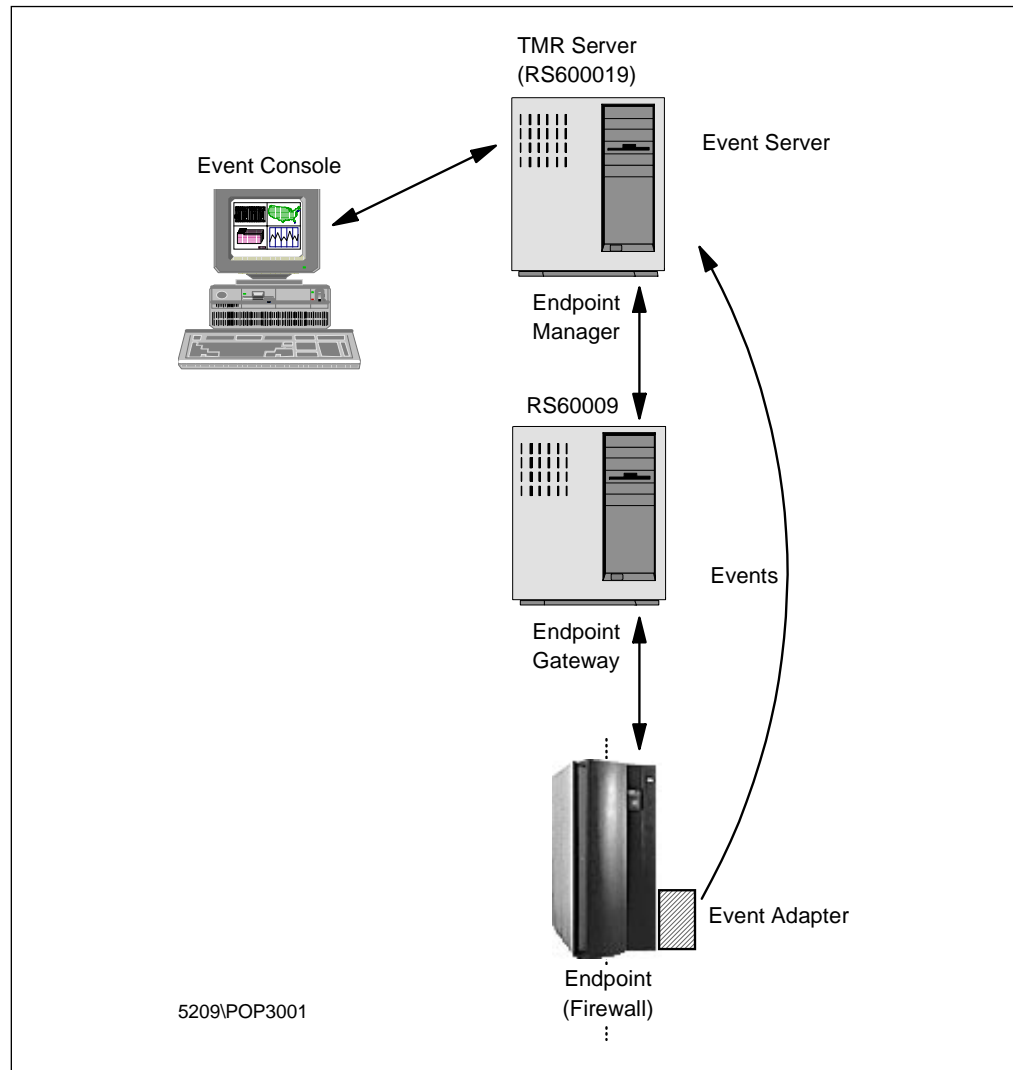


Figure 615. The Configuration of TME 10 for IBM eNetwork Firewall for Windows NT

- Endpoint Manager** The Endpoint Manager resides in the TMR Server and is created automatically. Its primary role is to assign endpoint to an endpoint gateway when the endpoint first logs in.
- Endpoint Gateway** The Endpoint Gateway performs all communications with its assigned endpoints without requiring additional communications with the TMR server. The gateway launches endpoint methods on the endpoints or runs gateway methods for the endpoints.
- Endpoint** An endpoint is a UNIX or PC workstation running the TME 10 endpoint service. The endpoints will send alerts to TME 10 Enterprise Console and execute tasks. The advantage of an endpoint is that we have the ability to fully manage the endpoint machine without requiring a large amount of disk space in the endpoint.
- Events** An event is the central unit of information within the TME 10 Enterprise Console. An event is any significant change in the status of a system resource or application.

Event Adapter	Event adapters are processes that typically reside on the same host as a managed source. A source is an application or a system resource. When an event adapter receives information from its source, the adapter formats the information and forwards it to the event server.
Event Server	An event server is a central server that handles all events in the distributed system. The event server creates an entry in a relational database for each incoming event. The event server then evaluates these events against a set of rules to determine if it can respond to or modify the event automatically.
Event Console	Event consoles provide a graphical user interface (GUI) that allows the IT staff to view and respond to dispatched events. Users can have an independent or shared view of events.

We will not cover the installation of TME 10 products in detail. We only discuss topics that are related with the firewall. For detail instructions on how to install and configure TME 10 you can refer to the TME 10 product manuals. The referenced books are listed below:

- *TME 10 Enterprise Console User's Guide Version 3.1*
- *TME 10 Enterprise Console Rule Builder's Guide Version 3.1*
- *TME 10 Enterprise Console Reference Manual Version 3.1*
- *TME 10 Enterprise Console Adapter's Guide Version 3.1*
- *TME 10 Cookbook for AIX*

Now, let's start with the installation of LCF and the Event Adapter.

21.6.1 Installation

The installation begins with putting LCF codes into the firewall. Since during the installation, LCF will broadcast UDP packets in order to search the LCF gateway, we have to enable traffic between the secure adapter of the firewall and the LCF gateway. The rules are shown below:

```
# Rules to enable traffic between Secure Adapter and LCF Gateway
permit s.s.s.1 0xffffffff L.C.F.G 0xffffffff any any 9494 any 9494 secure local outbound
permit L.C.F.G 0xffffffff s.s.s.1 0xffffffff any any 9494 any 9494 secure local inbound
```

Where s.s.s.1 is the IP address of the secure adapter of the firewall and L.C.F.G is the IP address of the LCF gateway. Port 9494 is the the default port number used by the endpoint and the endpoint gateway.

We have to install LCF for Windows NT locally since we are unable to install it remotely. At the time of writing this redbook, there is an an open APAR for this problem. Insert the CD-ROM of TME 10 Framework V3.2 into the firewall. Run SETUP from the directory \PC\LCF\WINNT. During the installation process, the setup program will ask for the port number that will be used to communicate between the firewall and the LCF gateway. The window is shown in Figure 616 on page 579.

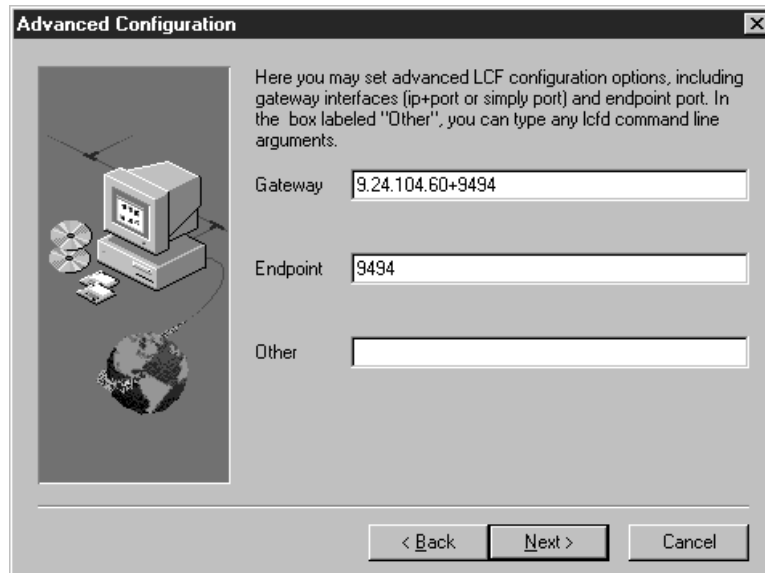


Figure 616. Advanced Configuration Window

We can enter the port number we want to use both in the firewall and the LCF gateway. The default port is 9494. To reduce broadcasts onto your network, we can enter the specific IP address of the LCF gateway. Enter the value 9.24.104.60+9494 in the first entry box as shown in Figure 616.

During the installation, the setup program will try to contact the LCF gateway. If it successfully contacted and logged into the LCF gateway, then the installation process of LCF is finished. If it hasn't, then we have to configure LAST.CFG in the directory `c:\tivoli\lcf\data\<your unique number>` (the unique number is the number that will be given by LCF) by appending the following line:

```
lcs.login_interfaces=9.24.104.60
```

Stop the LCFD service by issuing the command `net stop lcf` and start it again by using the command `net start lcf`.

In either case, you are now able to see through the TME 10 console that the firewall is now the member of the LCF gateway. To check it, open your TME 10 console, double-click on **Endpoint Manager** and then double-click on the Endpoint Gateway that owns the firewall. The window is shown in Figure 617 on page 580.



Figure 617. Endpoint List

Let's continue with event adapter installation.

Before we install the TME 10 Enterprise Console (TEC) Adapter for NT, we have to create rules in the firewall to enable the firewall communicate with the TEC server. The rules are shown below:

```
# Rules to enable traffic between Secure Adapter and TEC Server
permit s.s.s.1 0xffffffff T.E.C.S 0xffffffff tcp gt 1023 eq 1229 secure local outbound
permit T.E.C.S 0xffffffff s.s.s.1 0xffffffff tcp/ack eq 1229 gt 1023 secure local inbound
permit s.s.s.1 0xffffffff T.E.C.S 0xffffffff udp gt 1023 eq 111 secure local outbound
permit T.E.C.S 0xffffffff s.s.s.1 0xffffffff udp eq 111 gt 1023 secure local inbound
```

Where s.s.s.1 is the IP address of the secure adapter of the firewall and T.E.C.S is the IP address of the TEC server. See Chapter 8, "IBM Firewall Rule Base" on page 143 for configuring rules in the firewall.

The installation of TEC Adapter for NT is straight forward. Insert the CD-ROM, change to directory \ADAPTERS\W32-IX86\InstallNT and run SETUP. During the installation, the setup program will ask the name of the TEC Server where the events will be delivered, as shown below:



Figure 618. Server Configuration Window

We have to enter the name of our TEC Server. In our case, we enter rs600019 as our TEC Server. Please make sure that you can resolve the name of your TEC Server. To continue click on the **Next** button. After that, you will get a new service for the TEC adapter for NT. So, we'll have two more services for LCF **1** and TEC **2** as shown below:

```

EventLog
IBM Firewall Configuration Server
IBM Firewall Filter Log Service
IBM Firewall Log Daemon
IBM Firewall Proxy FTP Daemon
IBM Firewall Proxy Telnet Daemon
IBM Firewall SafeMail Server
IBM Firewall SOCKS Server
IBM Web Traffic Express
Microsoft DNS Server
NT LM Security Support Provider
Plug and Play
Remote Procedure Call (RPC) Service
Schedule
Server
TECNTAdapter 1
Tivoli Lightweight Client 2
Tivoli Remote Execution Service
Workstation

```

Now, we are ready to configure the TEC adapter files.

21.6.2 Configuration of TME 10 Enterprise Console

There are two steps that we need to do to set up our scenario. First, we will configure the event adapter to be able to send events to the event server and have the event server able to receive these events. Then we have to create rule sets (Tivoli rules, not the firewall's) to enable the event server to determine what to do with the incoming events. We'll start with the first step: configuring the event adapter.

There are three main files of the NT TEC adapter that are related with the configuration of IBM eNetwork Firewall for Windows NT integration. These are described below:

- tecad_nt.cfg** This is the global configuration file which defines the configuration options that can be used by the adapter.
- tecad_nt.cds** This is the Class Definition Statements (CDS) file which defines the event class definitions.
- tecad_nt.fmt** This is the format file which defines the format of system log messages and their mappings to classes.

These files are located in c:\tecnt\etc. Below are the steps to configure the TEC adapter for NT files.

1. Create a log file to monitor.

We have to create a log file from the firewall. Basically, what you have to do is to add one entry into Log Facilities from the IBM Firewall Configuration Client. Choose **Firewall Log** for the Facility, **Debug** for the Priority and the filename of the log for Log Filename. The other way is to use the existing log file. See 16.1, "Configure Logging" on page 409 for detail information about how to configure the log in IBM eNetwork Firewall for Windows NT.

2. Add lines into the tecad_nt.fmt file.

In order to recognize every line of the firewall log, we have to add lines into the format file of the adapter. The concept is we want to break the line into two types: fixed and variable. The variable parts are specified using the printf-type % specification. Currently, the following % specifications are supported:

- %s** Matches one string.
- %s*** Matches zero or more strings separated by white space.
- %s+** Matches one or more strings separated by white space.
- %t** Matches a time stamp of the following form:
month date time

For example, if we have a log file from the firewall as follows:

```
Mar 02 16:30:47 ntfirewall:1998;247: A4335;ICA2167i;andre;FTP Server;  
password;9.24.104.57;secure network;
```

Then, for specific message index ICA2167i the variable parts are:

- Mar 02 16:30:47
- ntfirewall:1998;247: A4325;
- andre
- FTP Server
- password
- 9.24.104.57
- secure network

Then, we can write a format string such as:

```
%t %s*;ICA2167i;%s;%s*;%s*;%s;%s*;
```


If you don't have any line in the result, then you may have an error in tecad_nt.fmt file. To check the error you can type the following command:

```
nt_gencds tecad_nt.fmt
```

Fix the error and run the first command again to create the tecad_nt.cds file. If you have successfully created the file, then the additional lines are shown in Figure 620.

```
CLASS FWNT_Login_Failed
  SELECT
    1: ATTR(=,"_event_id"), VALUE(=,"79");
    2: ATTR(=,"hostname");
    3: ATTR(=,"date");
    4: ATTR(=,"junk");
    5: ATTR(=,"userid");
    6: ATTR(=,"service");
    7: ATTR(=,"auth_scheme");
    8: ATTR(=,"client_addr");
    9: ATTR(=,"net_type");
  MAP
    hostname = $V2;
    origin = "9.24.104.57";
    date = $V3;
    msg_index= "2167";
    msg = PRINTF("%s authentication failed for %s using %s from %s on %s", $V5, $V6, $V7, $V8, $V9);
    msg_catalog = PRINTF("%s; %s; %s; %s; %s", $V5, $V6, $V7, $V9, $V8);
END
CLASS FWNT_Filterrules_Updated
  SELECT
    1: ATTR(=,"_event_id"), VALUE(=,"80");
    2: ATTR(=,"hostname");
    3: ATTR(=,"date");
    4: ATTR(=,"junk");
    5: ATTR(=,"rulestime");
    6: ATTR(=,"rulesdate");
  MAP
    hostname = $V2;
    origin = "9.24.104.57";
    date = $V3;
    msg_index = "1032";
    msg = PRINTF("Filter rules updated at %s on %s", $V5, $V6);
END
CLASS FWNT_Login_Succeeded
  SELECT
    1: ATTR(=,"_event_id"), VALUE(=,"81");
    2: ATTR(=,"hostname");
    3: ATTR(=,"date");
    4: ATTR(=,"junk");
    5: ATTR(=,"userid");
    6: ATTR(=,"service");
    7: ATTR(=,"auth_scheme");
    8: ATTR(=,"net_type");
    9: ATTR(=,"client_addr");
  MAP
    hostname = $V2;
    origin = "9.24.104.57";
    date = $V3;
    msg_index = "2169";
    msg = PRINTF("User %s successfully authenticated for %s using %s from %s:%s", $V5, $V6, $V7, $V8, $V9);
    msg_catalog = PRINTF("%s; %s; %s; %s; %s", $V5, $V6, $V7, $V9, $V8);
END
```

Figure 620. The Additional Lines of tecad_nt.cds

This is the class definition file that will be used by the TEC adapter to compose the event and match it with the BAROC file in the event server. First, the TEC

adapter will select the variables from the format file (SELECT attribute) and then map it into slot that will be recognized by the BAROC file (MAP attribute).

4. Change the global configuration file.

We have to add a few lines into `tecad_nt.conf` to tell the adapter which file will be polled. In our case, we want to point the TEC adapter to our firewall log. We've created a new firewall log and put it in the same directory as the TEC adapter configuration files. The additional lines are shown in Figure 621.

```
LogSources=ntfirewall.log
PollInterval=30
```

Figure 621. The Additional Lines of `tecad_nt.conf`

`LogSources` tells the adapter to poll the `ntfirewall.log` file. The working directory of the TEC adapter for NT is `c:\tecnt\etc`. `PollInterval` tells the adapter to poll the file every 30 seconds.

In order to put all changes into effect, we have to restart the TEC NT Adapter service. We can do it through the Windows NT Control Panel or using the command line interface:

```
net stop tecntadapter
```

To restart it, use the following command:

```
net start tecntadapter
```

If you want to debug the adapter, instead of using `net start tecntadapter` you can use the command below:

```
tecad_nt -d -c tecad_nt.conf
```

You will see a window that will give you all the information of what the adapter does. If any line of the firewall log matches with one of the formats from our format file, then the window will show you the variables that will be used to send the event.

5. Add a new BAROC file on the Event Server.

We are done with the configuration in the adapter site. Now, we will configure the event server to be able to receive the NT firewall events. What we should do is to create a new BAROC file. We go to the machine where the Event Server resides and create any file with extension BAROC. In our case, we create it in the `/usr/local/Tivoli/bin/generic/NTFirewall` directory and the file name is `tecad_nt_fw.baroc`. Add the following lines in the new file:

```

TEC_CLASSES :
    FWNT_Login_Failed ISA NT_Base
    DEFINES {
        severity: default="MINOR";
        sub_sources: default="FWNT";
    };
END

TEC_CLASSES :
    FWNT_Filterrules_Updated ISA NT_Base
    DEFINES {
        severity: default="MINOR";
        sub_sources: default="FWNT";
    };
END

TEC_CLASSES :
    FWNT_Login_Succeeded ISA NT_Base
    DEFINES {
        severity: default="HARMLESS";
        sub_sources: default="FWNT";
    };
END

```

Figure 622. A New BAROC File for NT Firewall

The names of the classes should match with the class names in the class definition file (CDS). Since we use the NT_Base class as our superclass, all slots in NT_Base are inherited to our new classes. That's why we don't have to define date and msg slots. We need sub_sources in order to filter the events and enable the Tivoli administrator to send all firewall-related events to the firewall administrator only.

6. Import, compile and load the new rule base classes.

If you have imported `tecad_nt_fw.baroc` before and you want to update it, you have to remove the rule base class from the existing rule base by entering the following command:

```
wdelrbclass tecad_nt_fw.baroc DM36
```

Where DM36 is our rule base that is currently running. First, we have to check the new BAROC file over the running rule base by issuing the following command:

```
wchkclass tecad_nt_fw.baroc DM36
```

Import the BAROC file into the existing rule base:

```
wimprbclass tecad_nt_fw.baroc DM36
```

Then, we have to compile the rule base since we added new classes:

```
wcomprules DM36
```

After we compile it, we have to reload the current rule base:

```
wloadrb DM36
```

Then to use the new rule base, we have to stop the event server and start it again by entering the following commands:

```
wstopesvr  
wstartesvr
```

We will check our new classes over the current rule base by entering the command:

```
wlsrbclass DM36 |grep FWNT
```

Then, we will see the list of our new classes. At this point, we can see events flowing from the firewall to the Event Server as shown in Figure 635 on page 595.

Now, we will create rule sets in the Event Server to enable the Event Server proactively doing something when it receives events.

The event server is the central server that processes information about events from the event adapter. The event server creates an entry for each incoming event and evaluates the event against a rule base to determine whether it can respond to or modify the event automatically.

A Rule is a description of any action that should be performed when a particular event is received by the event server. Rules are also used to correlate events to other events. A rule set contains a number of individual rules. A rule can specify the following typical actions:

- Modifications of the event's attributes.
- Dispatch of TME10 or other administrative actions.
- Deletion of the event
- Correlation to other events.
- Set a timer.
- Re-evaluation of a set of events.
- Prevention of display of duplicate events.

There are two types of rules: *simple* rules and *compound* rules. Simple rules pertain to only one event instance, however the rule may contain multiple event classes, each of which would cause evaluation of the rule. Compound rules allow us to specify a causal relationship between two event classes.

We create three rules to show the integration between TME 10 Enterprise Console and IBM eNetwork Firewall for Windows NT. The three rules are listed below:

- Send e-mail if FWNT_Filtrerrules_Updated arrives (simple rule).

This rule will send e-mail to the firewall administrator every time the IP filter rules in the firewall are re-activated.

- Close the FWNT_Login_Succeeded event within 10 minutes (simple rule).

This rule will remove the event that is generated by the firewall if somebody has successfully logged into the firewall. It will be removed after 10 minutes.

- Discard FWNT_Login_Failed if FWNT_Login_Succeeded arrives within 3 minutes (compound rule).

This rule will discard the event that is generated by the firewall if somebody has failed to log into the firewall and within 3 minutes he/she successfully logs into the firewall.

We only describe the first rule and the third rule since the second rule's type is the same as the first rule's. Let's start with the simple rule first.

1. Simple rule.

Open the Event Server Rule bases by double-clicking the **Event Server** icon from the Tivoli console. Select **Edit Rules...** from the pop-up menu of the rule base that you want to edit. After the TEC Rule Base dialog is displayed, select **New Ruleset** from the Rule Set menu. Then the TEC Rule Base dialog is displayed:

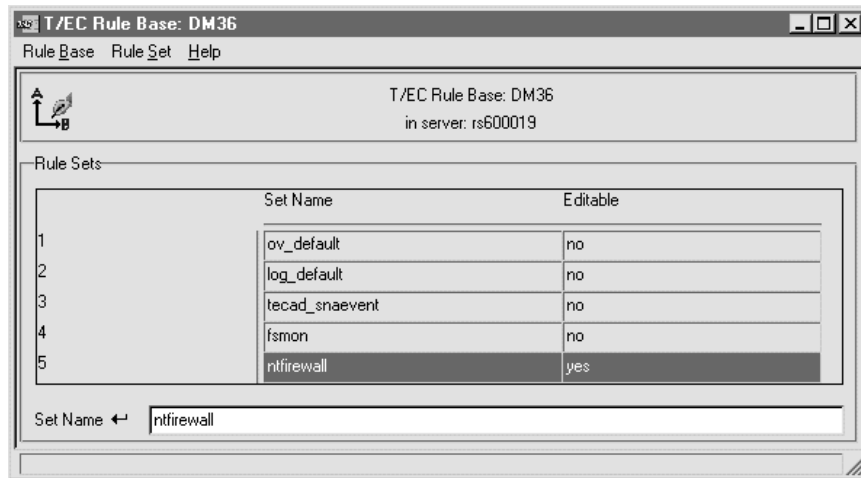


Figure 623. T/EC Rule Base Window

Enter the rule set name in the dialog box below. In our case we type ntfirewall and then press Enter. To create a simple rule, from the Ruleset dialog, select **Simple...** from the New Rule submenu of the Rule Set menu. And then we are shown a New Simple Rule window.

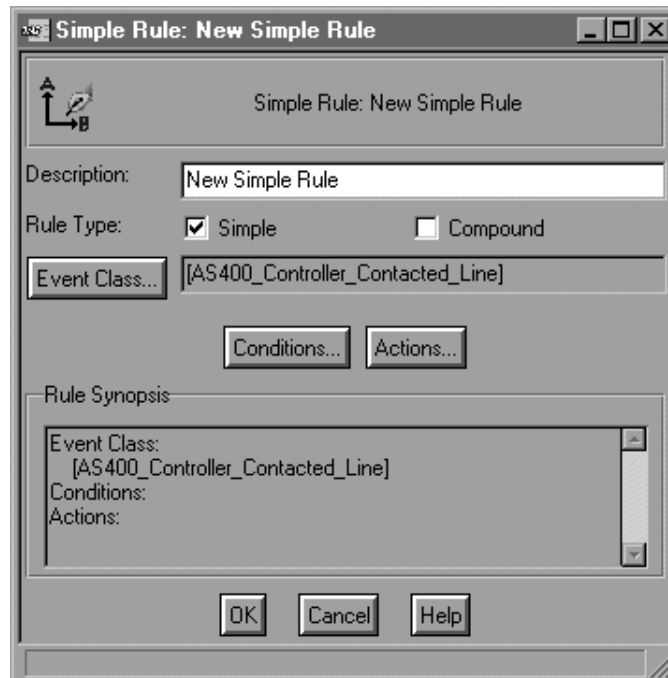


Figure 624. New Simple Rule Window

Type the description of the rule in the Description dialog box. Click on the **Event Class...** to choose the FWNT_Filterrules_Updated event class from the left box as shown in Figure 625 on page 589.

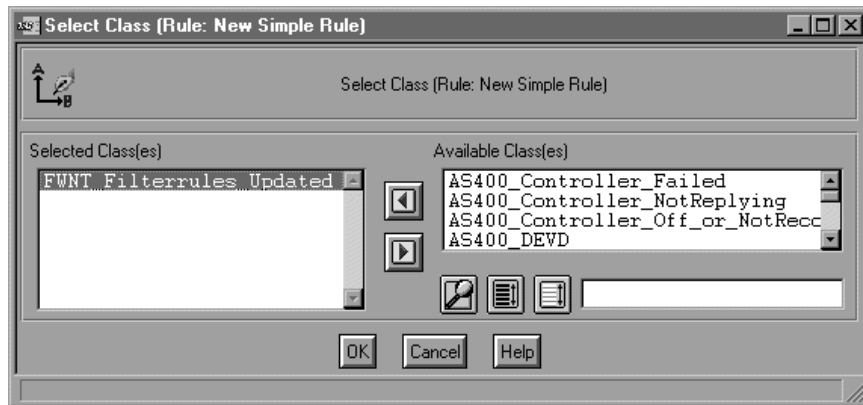


Figure 625. Select Event Class

Click on the **OK** button to save the changes. From the New Simple Rule window click the **Condition...** button to enter the rule's condition. Choose **origin** for Attribute, **in list** for Relation and enter the firewall IP address 9.24.104.57 for the Attribute Value(s) and press Enter. The complete window is shown below:

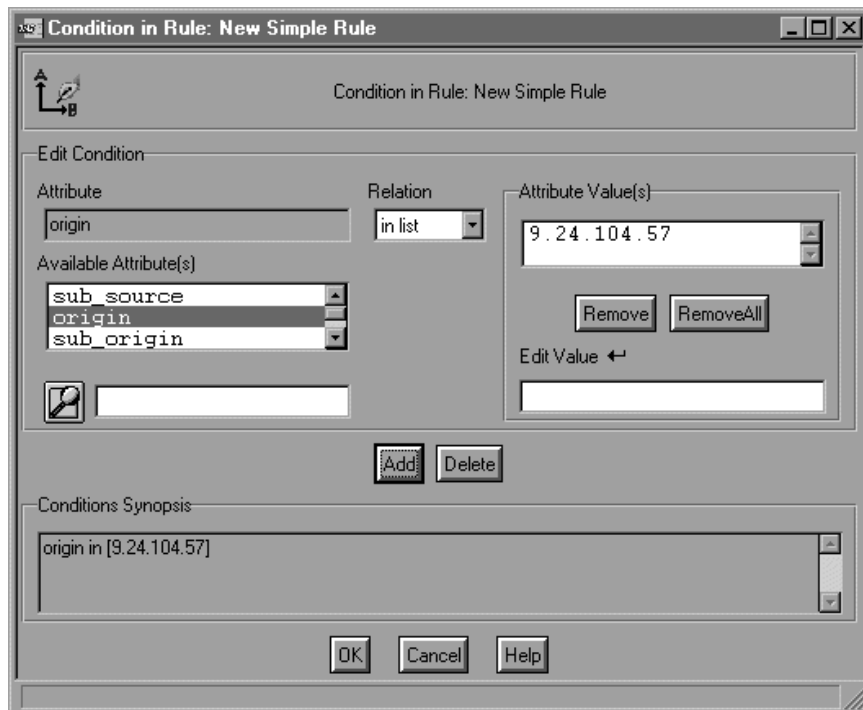


Figure 626. Condition in Rule Window

Click the **OK** button to save and go back to the New Simple Rule window. Then click on the **Actions...** button to choose the task that will be executed. Choose **When event is received** for When to Run and choose **Send Email** for Actions. Click on **Edit Arguments...** to enter the mail address of the firewall administrator. The completed window is shown in Figure 627 on page 590.

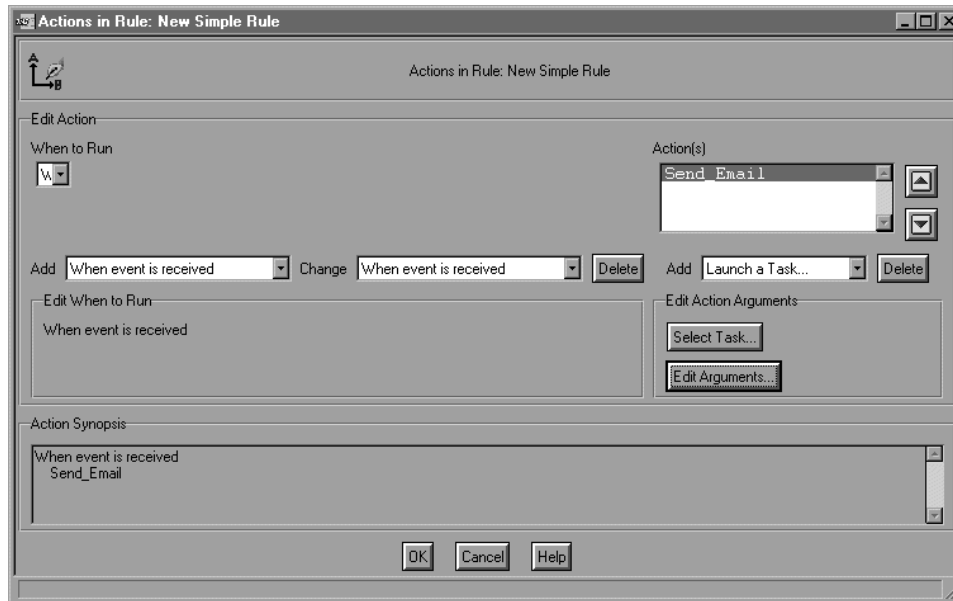


Figure 627. Action in Rule Window

Click the **OK** button to save the changes. The completed Simple Rule window is shown in Figure 628.

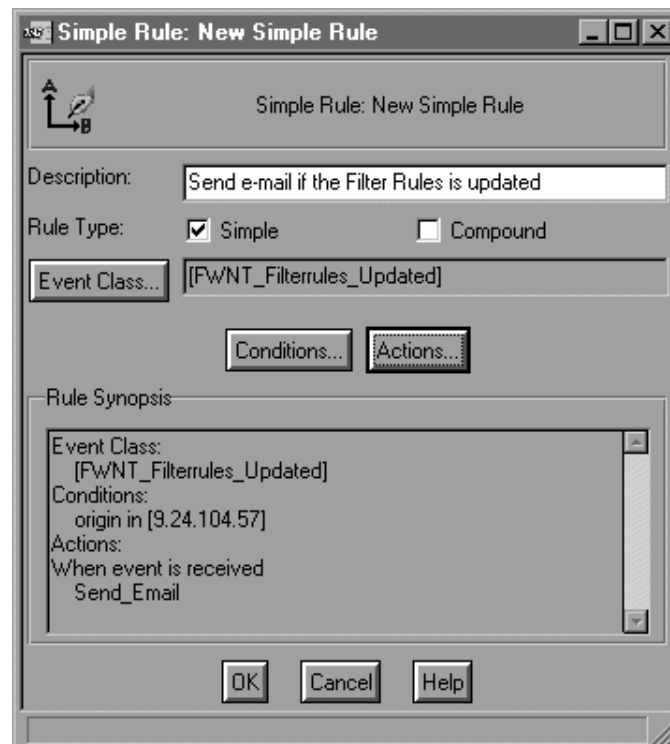


Figure 628. Completed Simple Rule Window

Click on the **OK** button to save the first simple rule.

2. Compound rule.

To create a compound rule, select **Compound...** from the New Rule submenu of the Rule Set menu. Then the Compound Rule: New Compound Rule dialog is displayed.

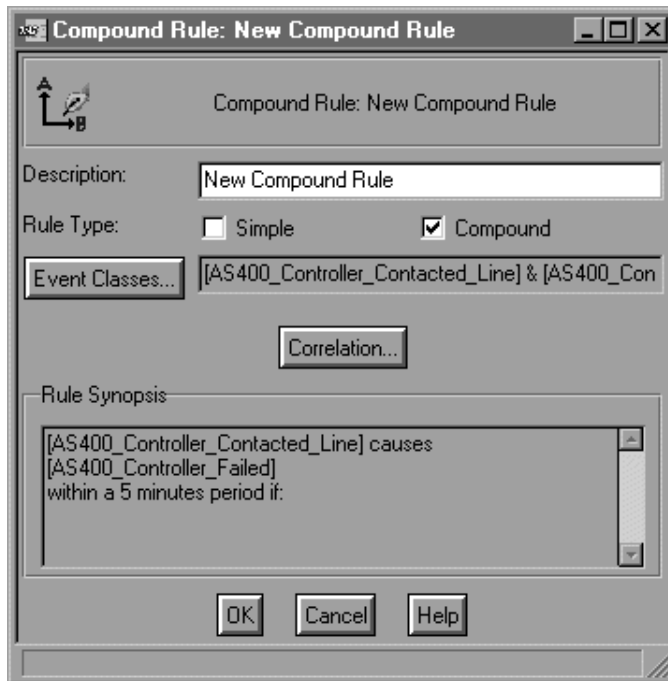


Figure 629. Compound Rule Window

Type the description of the new compound rule in the Description dialog box. Click on **Event Classes...** to choose the event classes that will be correlated. From the Select Class dialog, choose **FWNT_Login_Failed** and then **FWNT_Login_Succeeded**. The completed window is shown in Figure 630.

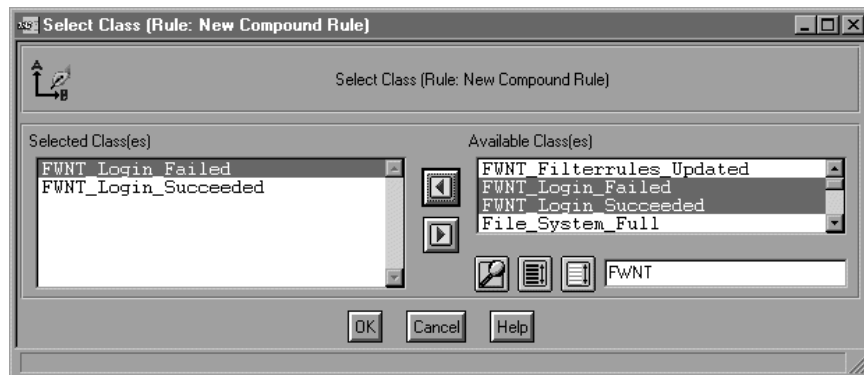


Figure 630. Completed Select Class Window for Compound Rule

Click on **OK** to save the changes and then from the Compound Rule: New Compound Rule dialog, click on **Correlation...** to choose the correlation for these two event classes above. When the Correlation in Rule window is displayed, choose **is canceled by** for the correlation type and **3** for the time. Choose **msg_catalog** from both event classes to specify the attribute that will be compared. And then choose **equals** for the Relation. The completed window of the correlation rule is displayed in Figure 631 on page 592.

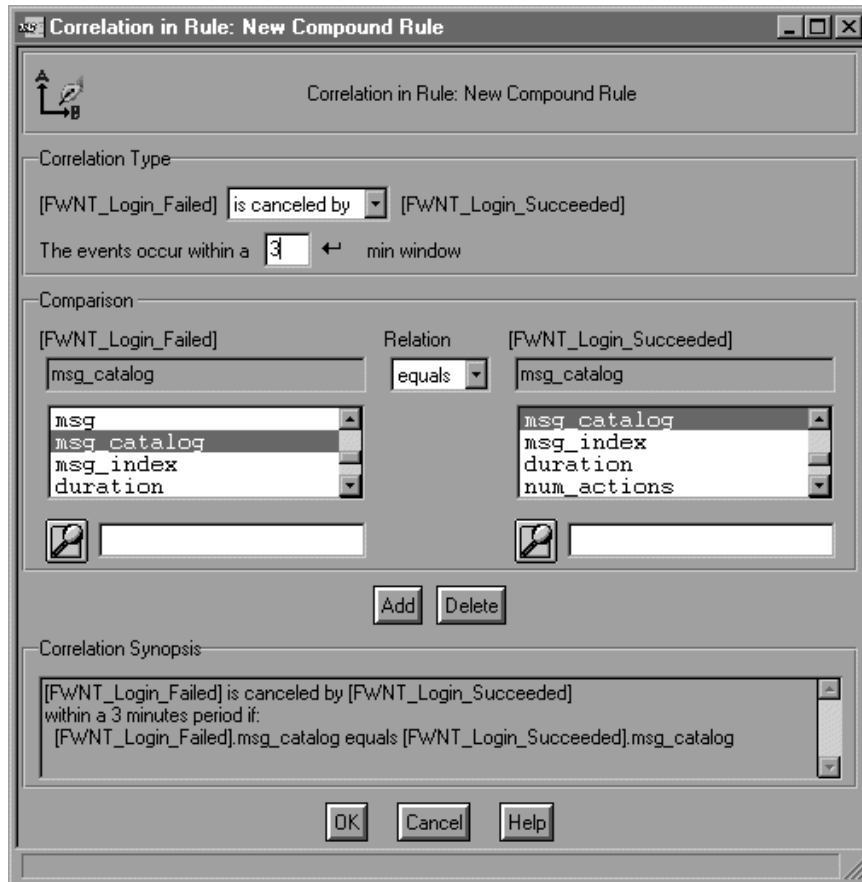


Figure 631. Correlation in Rule Window

Press the **Add** button to add the correlation and then press **OK** button to save the changes and go back to Compound Rule: New Compound Rule window.

The completed window of the new compound rule is shown in Figure 632 on page 593.

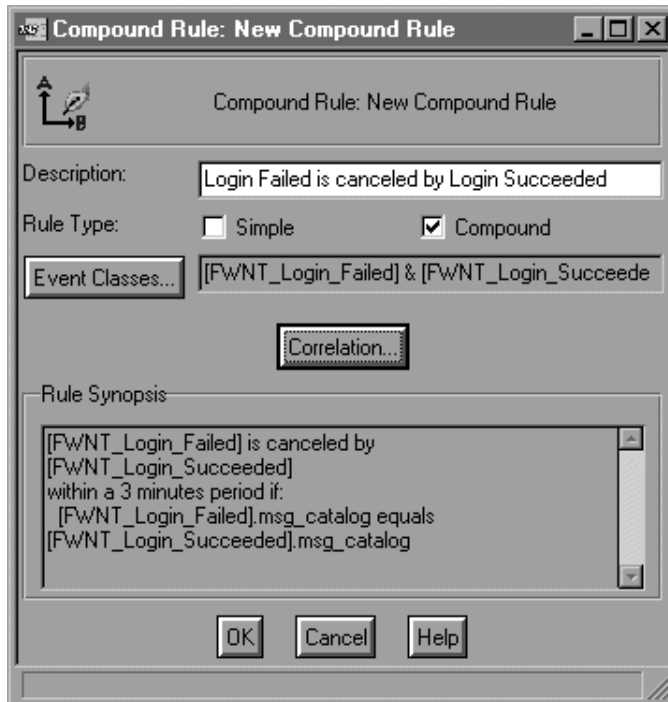


Figure 632. The Completed New Compound Rule

Click on the **OK** button to save the changes and now we can see in Figure 633 the two new rules that we've just created.

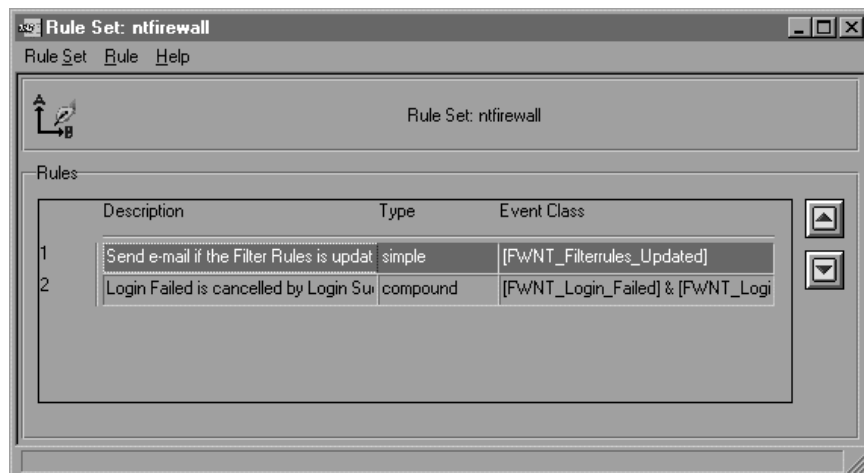


Figure 633. Two New Rules in the Rule Set

Select **Close** from the Rule Set pull-down menu in the Rule Set dialog. After the TEC Rule Base dialog for the rule base is displayed, then select **Save** from the pull-down menu of the Rule Base dialog to save the rule set file. After we save the rule set, we can close the dialog by selecting **Close** from the pull-down menu of the Rule Base dialog.

At this point, we have added two new rules in the new rule set. We have to do the same process when we import the new rule base class (tecad_nt_fw.baroc) as discussed above. Below is the process and the TEC commands to activate our new rule set:

1. Compile the rule base: wcomprules DM36.
2. Load the rule base: wloadrb DM36.
3. Stop the event server: wstopesvr.
4. Start the event server: wstartesvr.

Now, the rule engine on the event server contains our new rule set and we are ready to test our scenario.

21.6.3 Test the Configuration

We will test the incoming events from the NT event adapter to the event server. To monitor the firewall-related events only, we have to add a new Event Group into the event console (see *TME 10 Enterprise Console User's Guide*). After we have added a new event group for the firewall, open the event console by double-clicking on **Event Console** from TME 10 Desktop. The TME 10 Enterprise Console window is displayed (we only show you the Event Groups part of TEC Console) as shown in Figure 634.

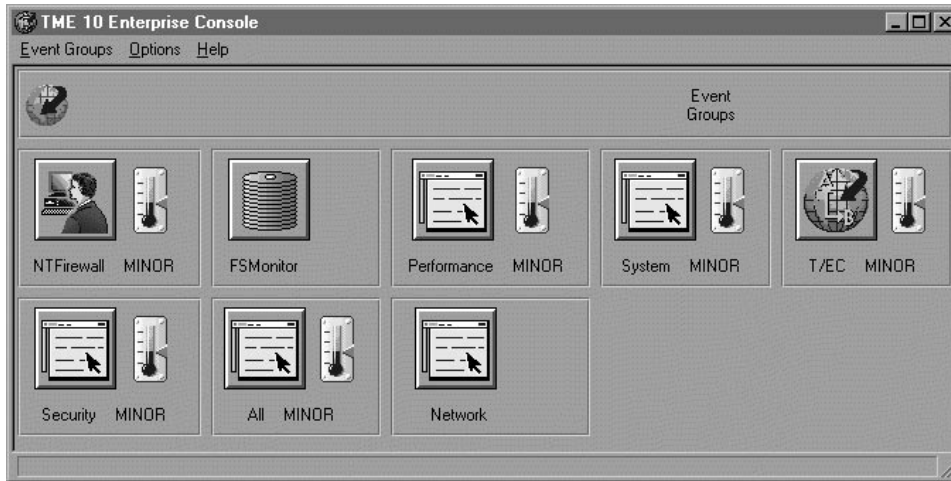


Figure 634. TME 10 Enterprise Console

Select the **NT Firewall** button to open the list of the incoming events. The window is shown in Figure 635 on page 595.

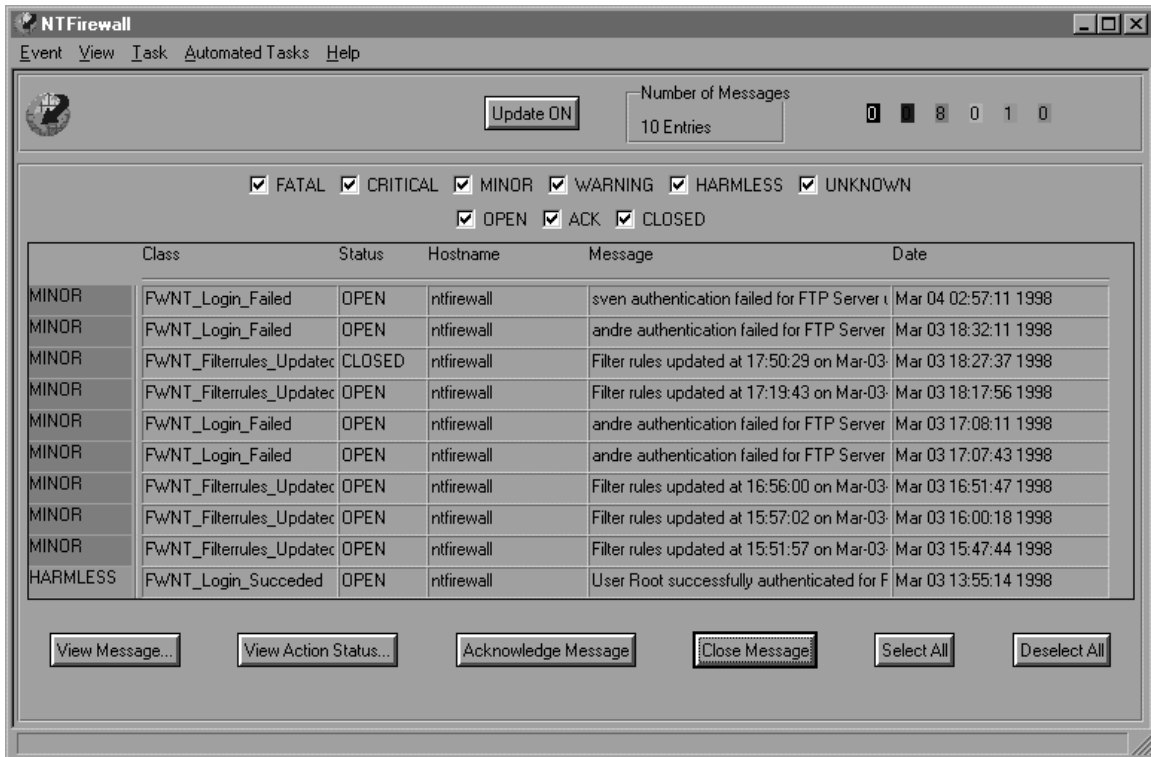


Figure 635. NTFirewall Event Window

From the dialog above, we can see events coming from the firewall regarding three conditions that we have defined before. To see detailed information of an event, double-click on the specific event and the Event Group Message Viewer is displayed:

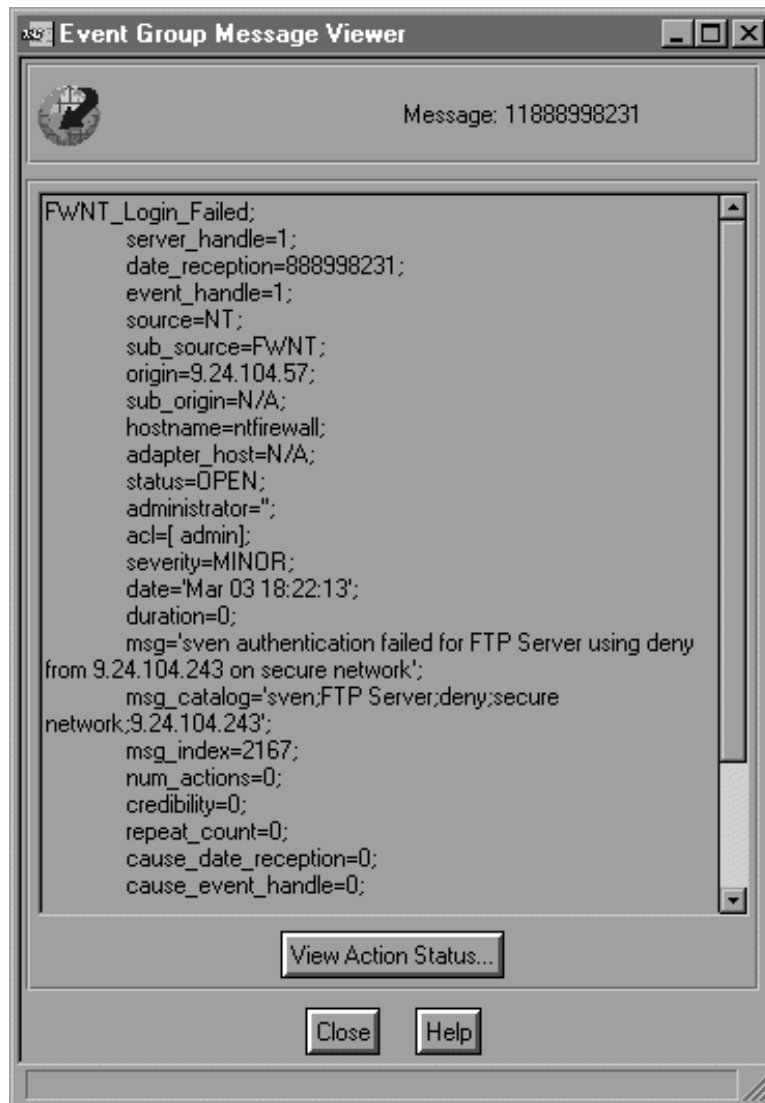


Figure 636. Event Group Message Viewer

After we apply the rules, the event server will not get the FWNT_Login_Failed event class if within 3 minutes FWNT_Login_Succeeded is triggered. And the FWNT_Login_Succeeded event class will be closed within 10 minutes because we don't need it. The Event server will send e-mail to the firewall administrator if the FWNT_Filterrules_Updated event class occurs.

Appendix A. Security Issues Old and New

Most firewall textbooks deal with network security issues in a host-based UNIX environment. While some of these concerns are still valid, we feel that the prevalence of Windows NT servers and Windows NT/Windows 95 clients in the enterprise puts new security issues at the forefront. For this reason, these new security issues are what we highlight in this chapter.

This chapter is divided into four main parts:

- Network vulnerability
- User and application vulnerability
- Windows NT vulnerability
- Traditional vulnerability

A.1 Network Vulnerability

We categorize under network vulnerability any attack which uses the network transport, that is, anything involving routers, the LAN infrastructure, and the network stacks of the different operating systems. We stop just below attacks involving applications and the unwitting (or knowing) cooperation of users.

A.1.1 Denial-of-Service Attacks

As the name suggests, a denial-of-service attack prevents you from using your computer or network resources. At best, the attacks represent an annoyance; at worst, they mean lost data and lost business. The main problem with denial-of-service attacks is that they are so easy to perform, and are therefore the preferred action of many cracker wannabees.

There are two general types of denial-of-service attacks. The first floods a target network with bogus traffic, eating into its bandwidth, and slowing down services. The second is directed at bringing down servers. Both exploit vulnerabilities in the operating systems of servers, clients, and networking equipment. Over the past two years, a number of denial-of-service attacks have been shown to be effective. Fortunately, vendors and user communities have been quite active in patching these vulnerabilities.

A well-known example of a denial-of-service attack is the ping of death. A ping of death is a ping packet of buffer size length 65527. This can cause the target computer system to crash. Discovered in 1996, the ping of death was shown to be effective against a number of operating systems. Patches have been issued to correct this.

Cousins of the ping of death are the teardrop attack, which uses corrupted UDP packets; the out-of-band attack, which uses a TCP packet of the same name; and bonk, a variant of the teardrop attack.

Vigilance is key in preventing denial-of-service attacks. Most denial-of-service attacks succeed because the target systems have not been patched by their administrators. To guard against this, you must maintain a close watch on security advisories from the Computer Emergency Response Team (CERT) through

<http://www.cert.org> and other security organizations. Watch also for patch releases from your vendors. A firewall reduces your exposure to denial-of-service attacks of this type because it prevents rogue packets from flowing into the secure packets. The firewall itself may go down, but your internal systems will be safe.

A.1.2 Snooping

Using tools like packet sniffers, crackers can read the messages going across your network. Consequently, a connection to the Internet without a firewall essentially throws your doors wide open to the world for this type of attack. At risk here is your data. Even worse, crackers could possibly read off your passwords.

Before, the favorite targets of snooping were Telnet, FTP, and mail. Now, we can add to the list the NetBIOS facility of Windows NT as well. NetBIOS was originally designed as a local area network protocol only with no routing capabilities. However, with NetBIOS over TCP/IP (also called NetBT), this limitation has been overcome, but new security problems have been introduced. Now, it is technically feasible to access Windows NT shared directories through the Internet!

Closely related to snooping is scanning. Scanning is done with software that methodically checks your computers' TCP/IP ports for services they may be running. This paves the way for attacks which are specific to a particular application, such as a Web server or a network file system (NFS) share. Chapter 18, "Testing The Configuration" on page 475 covers one such scanning tool which you can use to preempt crackers.

Chapter 18, "Testing The Configuration" on page 475 gives you a list of scanning tools available for Windows NT. It also shows you how you can use it to test the security of your firewall.

A firewall can be used to ward off any attempts at snooping and scanning on your internal network. This action covers the protocols we've discussed here. However, connections from your site going through the Internet to another site are still at risk; for protecting your traffic, encryption may be required.

A.1.3 Man-in-the-Middle

With a man-in-the-middle attack, a hostile computer intercepts traffic intended for another machine and impersonates it. This type of attack is so-called because the hostile computer places itself in the path of the two machines. UDP traffic is especially susceptible to this type of attack, so it's a good idea not to use UDP-dependent applications for any sensitive communication.

TCP is more resistant to man-in-the-middle because it uses sequence numbers; this is not to say that, given a sufficiently determined cracker, it cannot be broken.

However, to successfully pull off this kind of attack, the attacker must be quite close to one of the two endpoints (and not actually in the middle, as the name would suggest.) This is because the routing path becomes more complex the further the hostile user is from either end. Thus, to provide some bit of insurance against man-in-the-middle you must be reasonably sure of the network you are connecting to. This could mean selecting a trustworthy ISP.

The true insurance against man-in-the-middle is through the use of encryption and virtual-private network technology.

A.2 Application Vulnerability

Application vulnerability refers to weaknesses on the application that crackers can take advantage of. Given the size and complexity of today's computer programs, this is far more common than it was before. While firewalls can guard against these types of attacks, on the whole, they provide less coverage for application vulnerabilities, so you may need to take remedial steps elsewhere.

Some application vulnerabilities are discussed below.

A.2.1 Java, Javascript, and Active X

With the growing popularity and demand for ever-snazzier Web-based applications, manufacturers of Web browsers have souped up their products to such an extent that they are virtual operating systems in themselves. This means that malicious programmers can now write hostile programs that can do damage on the unfortunate user who downloads them. The rogue code could erase files on the client machine, or pilfer data to send back to its source server.

The three most popular Internet programming languages are Java, Javascript, and Active X. Of the three, the one with the worst reputation is Active X.

A firewall cannot really protect against these rogue programs, except when used to block Web traffic altogether. Filtering programs, such as Websweeper from Integralis, can work with the firewall to block these Internet programs, but they do not examine the code; it is an all-or-nothing proposition. Browsers also have options to turn off execution of code.

A.2.2 Viruses and Trojan Horses

Viruses and trojan horses ought to be a major concern because they have as much damage potential as a malicious hacker, and are far more common. But it is precisely because they are so common that they do not receive as much attention as the more spectacular exploits of crackers.

With the Internet, there are now more ways to transmit viruses and trojan horses. These could be downloaded from FTP and Web sites or attached as mail messages. (Didn't you run that nifty little game that your cousin sent you via e-mail yesterday? You know, the one where you shoot pies at a Certain Software Mogul from Redmond?) Worse still, there are macro viruses that execute from within spreadsheet and word-processing programs.

As with Internet program filters, there are also products that work with the firewall to scan data as they are downloaded from the Web or delivered by mail. See 21.4, "MIMESweeper" on page 550 to see one such product in action. This should not mean, however, that you can now forgo virus scanning on the desktop; it only means that you can put in place an additional layer of security. The proper place to do virus scanning is still on the desktop itself.

To secure against trojan horses, on the other hand, requires prudence on the part of users.

A.2.3 Dumb, Dumb Users!

And sometimes, it is the users themselves who pose a problem. This is the most difficult danger to guard against.

For example, how do you protect against users:

- Who disseminate trojan horse programs and virus-infected files?
- Who forward chain mail across the organization, to their relatives?
- Who unwittingly give out passwords and account information to a charming tech-support impersonator?
- Who delete all their files and their colleagues' files because they have been downsized?

These are situations in which the firewall is no defense. The only real solution here is a well-defined security policy and strong user education.

A.3 Windows NT Vulnerability

UNIX's weaknesses are well known, but only recently has Windows NT come under scrutiny for security. These investigations have shed light on some of its deficiencies. Apart from the network vulnerabilities mentioned in A.1, "Network Vulnerability" on page 597, there are some other details you have to watch out for.

- Windows NT File Access Control Lists. As shipped, Windows NT's file permissions are quite permissive as to what users can do. In principle, only administrators should have full control of the file system. Users should only have list permission, except for their own files.
- Network Access. By default, a Windows NT server gives everyone the right to log on and use its services. You should edit the user policies in NT's User Manager for Domains program to correct this.
- The Administrator Account. The administrator account is on every Windows NT machine, its name is well-known, and it doesn't lock if there are too many bad password attempts. A wise decision is to rename this account to a less-obvious name.
- Microsoft Internet Information Server. Microsoft IIS has had many problems. Earlier versions permitted users to execute commands on the Web server itself. Even now, the anonymous Internet user account also can have logon capabilities to the NT server.

Fortunately, several organizations keep track of the deficiencies in Windows NT and are slowly contributing to making it a stronger product. They also maintain Web sites from which you can keep yourself up-to-date. For a listing of recommended sites, please see H.4, "Web Sites" on page 624.

A.4 Traditional Vulnerabilities

Despite the changes in the landscape of the computer industry, it is still good practice to note the traditional vulnerabilities which have plagued computer systems. Some traditional vulnerabilities:

- Password failures. People still pick trivial passwords. At the same time, they write down in not-so-secret locations their more complex ones. You can use strong authentication systems that use one-time passwords to solve this.
- UDP-based applications. UDP does not provide the strong checking of TCP, so it is relatively easy to spoof them. Most applications that use UDP these days can be quite harmless, but some, such as NFS, can be real showstoppers. If you must run these applications, run them from behind a firewall and block all traffic of this type that reaches the IP filters.
- ICMP packets. The ICMP family, of which ping is a member, can do serious damage to older routers and operating systems. You can use the firewall to block them.
- DNS problems. DNS can be corrupted to assist in an attack, as is the case of older implementations of BIND, the DNS service daemon. Make sure you are running the latest version on your systems.
- Sendmail problems. Sendmail has been the point of entry for many hack attacks on UNIX systems. After the discovery of each vulnerability, a new release usually comes out (and it's for free). You should then take care to use only the latest versions.
- Telnet in cleartext. This problem relates to snooping, mentioned in A.1.2, "Snooping" on page 598. Telnet sessions are sent in cleartext and may be decoded by crackers. Where you need telnet across a non-secure network, you should probably use encryption.
- FTP problems. Some versions of FTP were a serious problem on UNIX systems, as crackers could use them to send in a file to give them logon access to a system. As with other applications, you should take care to use only the latest versions to circumvent this problem.
- Poorly written CGI Scripts. With poorly written CGI scripts, crackers could cause programs to execute on the Web server.

As a safety precaution, you should audit what systems you have and make sure that they do not carry any of the old vulnerabilities. Check regularly with your vendor or user group for updates. More traditional vulnerabilities are given in *Protect and Survive: Using IBM Firewall 3.1 for AIX*.

A.5 From Here On...

Security is a moving target, so to speak, because there's always something new to bug you. As a security administrator, you have a responsibility to keep yourself up-to-date on new developments. For more information, refer to the references and Web sites listed in Appendix H, "Related Publications, Web Sites, and Mailing Lists" on page 623.

Appendix C. Recommended ICA Message Tags

The following table lists recommended ICA message tags. This list is intended to help you to tailor the Log Monitor to your needs. You should at least create monitors for these tags, but check if all tags make sense in your environment. Check also the complete list of message tags in the *IBM eNetwork Firewall for Windows NT Reference Version 3 Release 3* to add additional message tags to monitor for.

The information can be used during creation of new message threshold monitors (see 16.2.5.2, "Configure Message Tag Monitors" on page 425).

Table 51 (Page 1 of 2). Recommended ICA Message Tags

Tag	Source	Comment
ICA0012	Log Monitor	Daemon received terminate signal.
ICA1004	Log Monitor	Filter logging daemon initialized.
ICA1005	Log Monitor	Suppressed logging due to buffer overflow.
ICA1006	Log Monitor	The fwlogd server failed in the indicated function, daemon terminated.
ICA1018	Log Monitor	The fwlogd server failed in the indicated function, daemon terminated.
ICA1019	Log Monitor	During startup of the filter logging daemon, the indicated system error was encountered.
ICA1020	Log Monitor	The fwlogd server failed in the indicated function, daemon terminated.
ICA1022	Log Monitor	Due to an error recorded prior to this message, filter support cannot be verified.
ICA1032	Log Monitor	IP packet filtering rules have been updated.
ICA1034	Log Monitor	Filter support deactivated. IP packet filtering now using default filter rules.
ICA1035	Log Monitor	Status of packet logging has changed.
ICA1044	Log Monitor	There are too many secure hosts try to connect with firewall machine.
ICA1045	Log Monitor	There are too many TCP sessions through firewall machine.
ICA1046	Log Monitor	There are too many UDP sessions through firewall machine.
ICA1070	VPN Tunnel	IPSec : Tunnel Authentication Header failure.
ICA1071	VPN Tunnel	IPSec : Encryption SP failure.
ICA1072	VPN Tunnel	IPSec : Tunnel expired.
ICA1200	Log Monitor	Terminating logging daemon due to errors
ICA1260	Log Monitor	The fwlogd daemon received the indicated termination signal and is stopping.
ICA1305	Log Monitor	In formatting an IP packet for syslog, a record was found with an unknown protocol specification.
ICA2006	Proxy	Indicated error returned by telnet audit process. Potential corruption of system files.
ICA2007	Proxy	Unknown error detected. Potential corruption of system files.
ICA2029	Proxy	The system call encounters a problem while executing a system call.
ICA2030	Proxy	The function call encounters a problem.
ICA2097	HTTP Proxy	Starting HTTP proxy server.
ICA2098	HTTP Proxy	Shutting down HTTP proxy server.
ICA2128	Proxy	A user attempted actions before entering in user name and password for authentication.
ICA2130	Proxy	Specified user attempted invalid command.
ICA2148	Proxy	Unauthorized connection attempt

Table 51 (Page 2 of 2). Recommended ICA Message Tags

Tag	Source	Comment
ICA2157	Proxy	Generally indicates an attempt to establish a connection to Firewall while transparent proxy is not configured.
ICA2168	Authentication	User ID could not be authenticated for service because there was a memory allocation failure during authentication processing.
ICA2171	Authentication	The password has expired and not been changed.
ICA2172	Authentication	This account has been locked.
ICA2173	Authentication	The ID supplied by the user is reserved for use by the firewall.
ICA2176	Authentication	Unknown authentication scheme.
ICA2180	SafeMail	SafeMail rejected session.
ICA2181	SafeMail	SafeMail rejected session for reason code.
ICA2182	SafeMail	SafeMail rejected session because of an invalid command.
ICA2183	HTTP Proxy	HTTP Proxy Configuration file is not available.
ICA2184	SafeMail	System error when SafeMail daemon attempted to establish signal handler.
ICA2185	SafeMail	Failure while opening the socket.
ICA2186	SafeMail	Failure while binding the socket to the port.
ICA2187	SafeMail	Failure while accepting new connection.
ICA2201	Proxy	The Networking component specified has failed.
ICA2203	Proxy	Memory error has occurred, usually out of memory
ICA2204	Proxy	The indicated service encountered an error when attempting to access or create the specified file or the file associated with the file parameter.
ICA2206	Proxy	The indicated configuration file is not in a usable format.
ICA3123	SOCKS	SOCKS server has successfully initialized and is awaiting connections.
ICA3127	SOCKS	SOCKS server received terminate signal.
ICA4023	Pager	Request to send page has failed.
ICA4033	Pager	Could not send the message onto the pager queue.
ICA4213	Pager	The paging queue is full.
ICA4363	Pager	Modem communication error.
ICA4364	Pager	Open com port error. Probably another program is using it.
ICA9003	Config Client	The firewall configuration server is unable to authenticate the indicated user.
ICA9011	Config Client	Configuration server not enabled for remote configuration.
ICA9013	Config Client	Remote configuration server received incorrect logon request.
ICA9014	Config Client	Remote configuration server unable to create pipe.
ICA9015	Config Client	Remote configuration server unable to create process.
ICA9037	Interfaces	Firewall interfaces being updated automatically.
ICA9038	Interfaces	Interface has been removed from Firewall configurations.
ICA9039	Interfaces	Interface has been added to the Firewall configuration.
ICA9040	Interfaces	Interface mask was updated.
ICA9041	Interfaces	No interfaces were found on this machine.
ICA9043	Network Address Translation	NAT failed to initialize and is inactive.
ICA9046	Network Address Translation	NAT has run out of ports with the many-to-one address.
ICA9050	Network Address Translation	NAT failed to translate a packet.

Appendix D. Protocol Numbers

The following table shows all defined protocol numbers from the RFC1700 with actual extensions. The RFC is available at <http://www.internic.net>.

The information can be used during creation of new rules (see Chapter 8, "IBM Firewall Rule Base" on page 143).

Table 52 (Page 1 of 3). Internet Protocol Numbers

Decimal	Keyword	Protocol	References
0	HOPOPT	IPv6 Hop-by-Hop Option	RFC1883
1	ICMP	Internet Control Message	RFC792, JBP
2	IGMP	Internet Group Management	RFC1112, JBP
3	GGP	Gateway-to-Gateway	RFC823, MB
4	IP	IP in IP (encapsulation)	JBP
5	ST	Stream	RFC1190, IEN119, JWF
6	TCP	Transmission Control	RFC793, JBP
7	UCL	UCL	PK
8	EGP	Exterior Gateway Protocol	RFC888, DLM1
9	IGP	any private interior gateway	JBP
10	BBN-RCC-MON	BBN RCC Monitoring	SGC
11	NVP-II	Network Voice Protocol	RFC741, SC3
12	PUP	PUP	PUP, XEROX
13	ARGUS	ARGUS	RWS4
14	EMCON	EMCON	BN7
15	XNET	Cross Net Debugger	IEN158, JFH2
16	CHAOS	Chaos	NC3
17	UDP	User Datagram	RFC768, JBP
18	MUX	Multiplexing	IEN90, JBP
19	DCN-MEAS	DCN Measurement Subsystems	DLM1
20	HMP	Host Monitoring	RFC869, RH6
21	PRM	Packet Radio Measurement	ZSU
22	XNS-IDP	XEROX NS IDP	ETHERNET, XEROX
23	TRUNK-1	Trunk-1	BWB6
24	TRUNK-2	Trunk-2	BWB6
25	LEAF-1	Leaf-1	BWB6
26	LEAF-2	Leaf-2	BWB6
27	RDP	Reliable Data Protocol	RFC908, RH6
28	IRTP	Internet Reliable Transaction	RFC938, TXM
29	ISO-TP4	ISO Transport Protocol Class 4	RFC905, RC77
30	NETBLT	Bulk Data Transfer Protocol	RFC969, DDC1
31	MFE-NSP	MFE Network Services Protocol	MFENET, BCH2
32	MERIT-INP	MERIT Internodal Protocol	HWB
33	SEP	Sequential Exchange Protocol	JC120
34	3PC	Third Party Connect Protocol	SAF3
35	IDPR	Inter-Domain Policy Routing Protocol	MXS1
36	XTP	XTP	GXC
37	DDP	Datagram Delivery Protocol	WXC
38	IDPR-CMTP	IDPR Control Message Transport Protocol	MXS1
39	TP++	TP++ Transport Protocol	DXF
40	IL	IL Transport Protocol	DXP2
41	SIP	Simple Internet Protocol	SXD
42	SDRP	Source Demand Routing Protocol	DXE1
43	SIP-SR	SIP Source Route	SXD
44	SIP-FRAG	SIP Fragment	SXD
45	IDRP	Inter-Domain Routing Protocol	Sue Hares

Table 52 (Page 2 of 3). Internet Protocol Numbers

Decimal	Keyword	Protocol	References
46	RSVP	Reservation Protocol	Bob Braden
47	GRE	General Routing Encapsulation	Tony Li
48	MHRP	Mobile Host Routing Protocol	David Johnson
49	BNA	BNA	Gary Salamon
50	SIPP-ESP	SIPP Encap Security Payload	Steve Deering
51	SIPP-AH	SIPP Authentication Header	Steve Deering
52	I-NLSP	Integrated Net Layer Security TUBA	GLENN
53	SWIPE	IP with Encryption	JI6
54	NHRP	NBMA Next Hop Resolution Protocol	RFC1735
55	MOBILE	IP Mobility	Perkins
56	TLSP	Transport Layer Security Protocol using Kryptonnet key management	Oberg
57	SKIP	SKIP	Markson
58	IPv6-ICMP	ICMP for IPv6	RFC1883
59	IPv6-NoNxt	No Next Header for IPv6	RFC1883
60	IPv6-Opts	Destination Options for IPv6	RFC1883
61		any host internal protocol	JBP
62	CFTP	CFTP	CFTP, HCF2
63		any local network	JBP
64	SAT-EXPAK	SATNET and Backroom EXPAK	SHB
65	KRYPTOLAN	Kryptolan	PXL1
66	RVD	MIT Remote Virtual Disk Protocol	MBG
67	IPPC	Internet Pluribus Packet Core	SHB
68		any distributed file system	JBP
69	SAT-MON	SATNET Monitoring	SHB
70	VISA	VISA Protocol	GXT1
71	IPCV	Internet Packet Core Utility	SHB
72	CPNX	Computer Protocol Network Executive	DXM2
73	CPHB	Computer Protocol Heart Beat	DXM2
74	WSN	Wang Span Network	VXD
75	PVP	Packet Video Protocol	SC3
76	BR-SAT-MON	Backroom SATNET Monitoring	SHB
77	SUN-ND	SUN ND PROTOCOL-Temporary	WM3
78	WB-MON	WIDEBAND Monitoring	SHB
79	WB-EXPAK	WIDEBAND EXPAK	SHB
80	ISO-IP	ISO Internet Protocol	MTR
81	VMTP	VMTP	DRC3
82	SECURE-VMTP	SECURE-VMTP	DRC3
83	VINES	VINES	BXH
84	TTP	TTP	JXS
85	NSFNET-IGP	NSFNET-IGP	HWB
86	DGP	Dissimilar Gateway Protocol	DGP, ML109
87	TCF	TCF	GAL5
88	IGRP	IGRP	CISCO, GXS
89	OSPFIGP	OSPFIGP	RFC1583, JTM4
90	Sprite-RPC	Sprite RPC Protocol	SPRITE, BXW
91	LARP	Locus Address Resolution Protocol	BXH
92	MTP	Multicast Transport Protocol	SXA
93	AX.25	AX.25 Frames	BK29
94	IPIP	IP-within-IP Encapsulation Protocol	JI6
95	MICP	Mobile Internetworking Control Pro.	JI6
96	SCC-SP	Semaphore Communications Sec. Pro.	HXH
97	ETHERIP	Ethernet-within-IP Encapsulation	RXH1
98	ENCAP	Encapsulation Header	RFC1241, RXB3
99		any private encryption scheme	JBP
100	GMTP	GMTP	RXB5

Table 52 (Page 3 of 3). Internet Protocol Numbers

Decimal	Keyword	Protocol	References
101	IFMP	Ipsilon Flow Management Protocol	Hinden
102	PNNI	PNNI over IP	Callon
103	PIM	Protocol Independent Multicast	Farinacci
104	ARIS	ARIS	Feldman
105	SCPS	SCPS	Durst
106	QNX	QNX	Hunter
107	A/N	Active Networks	Braden
108	IPPCP	IP Payload Compression Protocol	Doraswamy
109	SNP	Sitara Networks Protocol	Sridhar
110	Compaq-Peer	Compaq Peer Protocol	Volpe
111	IPX-in-IP	IPX in IP	Lee
112	VRRP	Virtual Router Redundancy Protocol	Hinden
113	PGM	PGM Reliable Transport Protocol	Speakman
114-254		Unassigned	JBP
255		Reserved	JBP

Appendix E. Well-Known Port Numbers

The following table shows some of the more common IP well-known port numbers. The port numbers are divided into three ranges:

- Well-Known Ports from 0 through 1023.
- Registered Ports from 1024 through 49151.
- Dynamic and/or Private Ports from 49152 through 65535.

The official list of all assigned numbers is maintained in an RFC. The latest version is RFC1700, which obsoletes the previous list, RFC1340. The RFC is available at <http://www.internic.net>.

Table 53 (Page 1 of 2). Internet Port Numbers

Keyword	Decimal/protocol	Description
echo	7/tcp, 7/udp	
discard	9/tcp, 9/udp	Sink null
sysstat	11/tcp	Active users information
daytime	13/tcp, 13/udp	
qotd	17/tcp	Quote of the Day
chargen	19/tcp, 19/udp	Character generator
ftp-data	20/tcp	File Transfer Protocol (data)
ftp	21/tcp	File Transfer Protocol (control)
telnet	23/tcp	Telnet
smtp	25/tcp	Simple Mail Transfer Protocol
time	37/tcp, 37/udp	Time server
rlp	39/tcp, 39/udp	Resource Location Protocol
nicname	43/tcp	Who is
domain	53/tcp, 53/udp	Domain Name Server
sql*net	66/tcp, 66/udp	Oracle SQL*NET
bootps	67/udp	Bootstrap Protocol server
bootpc	68/udp	Bootstrap Protocol client
tftp	69/udp	Trivial File Transfer Protocol
gopher	70/tcp	Gopher
finger	79/tcp	Finger information system
http, www-http	80/tcp	Hyper Text Transfer Protocol, World Wide Web HTTP
kerberos	88/tcp	Kerberos security system
npp	92/tcp	Network Printing Protocol
hostname	101/tcp	NIC host name server
pop2	109/tcp	Post Office Protocol version 2
pop3	110/tcp	Post Office Protocol version 3
sunrpc	111/tcp, 111/udp	SUN Remote Procedure Call
auth, ident	113/tcp	Authentication service, ident service
sftp	115/tcp	Simple File Transfer Protocol
uucp-path	117/tcp	UUCP path service
nntp	119/tcp	Network News Transfer Protocol
ntp	123/tcp, 123/udp	Network Time Protocol
cisco-xxx	130-132	Various Cisco-specific protocols
ingres-net	134/tcp	Ingres-net service
rpc-mapper	135/tcp	Port Mapper used for Microsoft Exchange
netbios-ns, nbname	137/tcp	NetBIOS name service
netbios-dgm, nbdatagram	138/tcp	NetBIOS datagram service

Table 53 (Page 2 of 2). Internet Port Numbers

Keyword	Decimal/protocol	Description
netbios-ssn, nbssession	139/tcp	NetBIOS session service
imap	143/tcp	Internet Message Access Protocol
snmp	161/udp	SNMP gets and sets
snmptrap	162/udp	SNMP traps
xdmcp	177/tcp	X Display Manager Control Protocol
irc	194/tcp	Internet Relay Chat
wais	210/tcp	Wide Area Information System
set	257/tcp	Secure Electronic Transaction
netware-ip	396/udp	Novell Netware over IP
hyper-g	418/tcp	Hyper-G
https	443/tcp	HTTP over SSL
exec	512/tcp	Remote Command Execution (rexec)
biff	512/udp	Inform users of new mail received
login	513/tcp	Remote login (rlogin)
who	513/udp	Who is logged on
shell	514/tcp	Remote Shell (rsh)
syslog	514/udp	UNIX logging port
printer	515/tcp	Print spooler
talk	517/udp	Interactive messaging
timed	525/udp	timeserver
uucp	540/tcp	UNIX-to-UNIX copy program
nntp	563/tcp	NNTP over SSL
netviewdm1	729/tcp	IBM NetView Distribution Manager server/client
netviewdm2	730/tcp	IBM NetView Distribution Manager send
netviewdm3	731/tcp	IBM NetView Distribution Manager receive
fwconfig	1014/tcp	IBM Firewall Configuration Client
exch-is	1039/tcp	Information Store used for Microsoft Exchange
exch-ds	1039/tcp	Directory Store used for Microsoft Exchange
SOCKS	1080/tcp	SOCKS application-level gateway
lotusnote	1352/tcp	Lotus Notes
wins	1512/tcp	MS Windows Internet Name Service
nfs	2049/udp	SUN Network File System
sap	3200-3499/tcp	SAP R/3
smt	5051/tcp	SOCKS Monitor Traffic
x11	6000-6063/tcp	X Windows system
real-stream	6970-7170/udp	Real Audio Stream
realaudio	7070/tcp	Real Audio Connection
hod	8999/tcp	Host on Demand Connection
lcf	9494/tcp	Tivoli Light Weight Client Framework Connection
hod redirector	12001/tcp	Host on Demand Redirector Connection

Appendix F. ICMP Message Types and Codes

The following table shows a list of all ICMP message types and the RFCs in which they are defined, taken from the latest RFC of Assigned Numbers (RFC 1700) with actual extensions. The RFC can be obtained from <http://www.internic.net>.

Type	Name	Reference
0	Echo Reply	RFC792
1	Unassigned	JBP
2	Unassigned	JBP
3	Destination Unreachable	RFC792
4	Source Quench	RFC792
5	Redirect	RFC792
6	Alternate Host Address	JBP
7	Unassigned	JBP
8	Echo	RFC792
9	Router Advertisement	RFC1256
10	Router Selection	RFC1256
11	Time Exceeded	RFC792
12	Parameter Problem	RFC792
13	Timestamp	RFC792
14	Timestamp Reply	RFC792
15	Information Request	RFC792
16	Information Reply	RFC792
17	Address Mask Request	RFC950
18	Address Mask Reply	RFC950
19	Reserved (for Security)	Solo
20-29	Reserved (for Robustness Experiment)	ZSu
30	Traceroute	RFC1393
31	Datagram Conversion Error	RFC1475
32	Mobile Host Redirect	David Johnson
33	IPv6 Where-Are-You	Bill Simpson
34	IPv6 I-Am-Here	Bill Simpson
35	Mobile Registration Request	Bill Simpson
36	Mobile Registration Reply	Bill Simpson
37	Domain Name Request	RFC1788
38	Domain Name Reply	RFC1788
39	SKIP	Markson
40	Photuris	Bill Simpson
41-255	Reserved	JBP

The following is the list of the code numbers, taken also from the same RFC.

Type	Name	Reference
0	Echo Reply Codes 0 No Code	RFC792
1	Unassigned	JBP
2	Unassigned	JBP
3	Destination Unreachable Codes 0 Net Unreachable	RFC792

Table 55 (Page 2 of 3). ICMP Message Codes

Type	Name	Reference
	1 Host Unreachable	
	2 Protocol Unreachable	
	3 Port Unreachable	
	4 Fragmentation Needed and Don't Fragment was Set	
	5 Source Route Failed	
	6 Destination Network Unknown	
	7 Destination Host Unknown	
	8 Source Host Isolated	
	9 Communication with Destination Network is Administratively Prohibited	
	10 Communication with Destination Host is Administratively Prohibited	
	11 Destination Network Unreachable for Type of Service	
	12 Destination Host Unreachable for Type of Service	
	13 Communication Administratively Prohibited by Filtering	RFC1812
	14 Host Precedence Violation	RFC1812
	15 Precedence Cutoff in Effect	RFC1812
4	Source Quench	RFC792
	Codes	
	0 No Code	
5	Redirect	RFC792
	Codes	
	0 Redirect Datagram for the Network (or subnet)	
	1 Redirect Datagram for the Host	
	2 Redirect Datagram for the Type of Service and Network	
	3 Redirect Datagram for the Type of Service and Host	
6	Alternate Host Address	JBP
	Codes	
	0 Alternate Address for Host	
7	Unassigned	JBP
8	Echo	RFC792
	Codes	
	0 No Code	
9	Router Advertisement	RFC1256
	Codes	
	0 No Code	
10	Router Selection	RFC1256
	Codes	
	0 No Code	
11	Time Exceeded	RFC792
	Codes	
	0 Time to Live exceeded in Transit	
	1 Fragment Reassembly Time Exceeded	
12	Parameter Problem	RFC792
	Codes	
	0 Pointer Indicates the Error	
	1 Missing a Required Option	RFC1108
	2 Bad Length	
13	Timestamp	RFC792
	Codes	
	0 No Code	
14	Timestamp Reply	RFC792
	Codes	
	0 No Code	
15	Information Request	RFC792
	Codes	

Table 55 (Page 3 of 3). ICMP Message Codes

Type	Name	Reference
16	0 No Code Information Reply Codes	RFC792
17	0 No Code Address Mask Request Codes	RFC950
18	0 No Code Address Mask Reply Codes	RFC950
19	0 No Code Reserved (for Security)	Solo
20-29	Reserved (for Robustness Experiment)	ZSu
30	Traceroute	RFC1393
31	Datagram Conversion Error	RFC1475
32	Mobile Host Redirect	David Johnson
33	IPv6 Where-Are-You	Bill Simpson
34	IPv6 I-Am-Here	Bill Simpson
35	Mobile Registration Request	Bill Simpson
36	Mobile Registration Reply	Bill Simpson
37	Domain Name Request	RFC1788
38	Domain Name Reply Codes	RFC1788
39	0 No Code SKIP	Markson
40	Photuris Codes	Bill Simpson
	0 Reserved	
	1 Unknown Security Parameters Index	
	2 Valid Security Parameters, but Authentication Failed	
	3 Valid Security Parameters, but Decryption Failed	

Appendix G. List of IBM Firewall Rules

We created a QBASIC program to produce a list of IBM eNetwork Firewall for Windows NT rules. We used QBASIC since it is provided by Windows NT. To run the program, enter the following command in the C:\> prompt:

```
QBASIC /RUN fwrules.bas
```

If you want to change the destination and the directory of the output file, then you can change manually the variable outfile\$ from the code.

Below is the listing of fwrules.bas:

```
REM *****
REM *** Declarations ***
REM *****

DECLARE FUNCTION GetSign$ (letsign$)
DECLARE FUNCTION GetFlag$ (flag$)
DECLARE SUB GetWord (rulesdata$)

rulesfile$ = "c:\program files\ibm\firewall\config\fwrules.cfg"
outfile$ = "c:\temp\fwrules.out"

DIM SHARED word$(15)

REM *****
REM *** Main Program ***
REM *****

OPEN outfile$ FOR OUTPUT AS #1
  OPEN rulesfile$ FOR INPUT AS #2
  DO
    LINE INPUT #2, rulesdata$

    GetWord (rulesdata$)

    index$ = "Rules No.: " + word$(1)
    title$ = word$(2) + SPACE$(10) + word$(3)
    subtitle$ = word$(4) + " " + word$(12) + " direction(s) to " + word$(10) + " interface(s)"

    IF word$(5) = "tcp/ack" THEN
      destproc$ = "tcp"
    ELSE
      destproc$ = word$(5)
    END IF

    from$ = "From: " + word$(5) + " " + GetSign$(word$(6)) + " " + word$(7)
    to$ = "To: " + destproc$ + " " + GetSign$(word$(8)) + " " + word$(9)
    connection$ = from$ + to$

    logfragment$ = "Log = " + GetFlag$(word$(13)) + SPACE$(5) + "Fragment = " + GetFlag$(word$(14))

    PRINT #1, index$
    PRINT #1, title$
    PRINT #1, subtitle$
    PRINT #1, connection$
    PRINT #1, logfragment$
    PRINT #1, ""

  LOOP UNTIL (EOF(2))
  CLOSE #2
CLOSE #1

FUNCTION GetFlag$ (flag$) STATIC
  IF RIGHT$(flag$, 1) = "n" THEN
    GetFlag$ = "No"
  ELSE
```

```

        GetFlag$ = "Yes"
    END IF
END FUNCTION

FUNCTION GetSign$ (letsign$) STATIC
    listletsign$ = "gtlteq"
    listsign$ = "><="
    IF INSTR(listletsign$, letsign$) = 0 THEN
        GetSign$ = letsign$
    ELSE
        GetSign$ = MID$(listsign$, INT(INSTR(listletsign$, letsign$) / 2) + 1, 1)
    END IF
END FUNCTION

SUB GetWord (rulesdata$)
    i = 1
    DO WHILE INSTR(rulesdata$, "|") <> 0
        length = LEN(rulesdata$)
        location = INSTR(rulesdata$, "|")
        word$(i) = LEFT$(rulesdata$, location - 1)
        rulesdata$ = RIGHT$(rulesdata$, length - location)
        i = i + 1
    LOOP
END SUB

```

Some of the contents of the output file are shown below:

```

Rules No.: 502
Deny DMZ          deny All DMZ
deny inbound direction(s) to specific(DMZ) interface(s)
From: all any 0 To: all any 0
Log = Yes      Fragment = Yes

```

```

Rules No.: 70
DNS - Long Client Queries          TCP port 53
permit both direction(s) to both interface(s)
From: tcp > 1023 To: tcp = 53
Log = No      Fragment = Yes

```

```

Rules No.: 31
DNS - Long Replies          TCP/ACK port 53
permit both direction(s) to both interface(s)
From: tcp/ack = 53 To: tcp > 1023
Log = No      Fragment = Yes

```

```

Rules No.: 52
DNS - Long Server Queries          TCP port 53
permit both direction(s) to both interface(s)
From: tcp = 53 To: tcp = 53
Log = No      Fragment = Yes

```

You can download this code as detailed in "How to Access Samples from This Book" on page xiii; the name of the file is fwrules.bas.

We also coded a REXX program that lists the services and captures the rules specified. This program reads each service defined in c:\Program Files\IBM\Firewall\config\fwservices.cfg; the 8th parameter in each record points to the rule(s) used for that service. The program then reads the rule from c:\Program Files\IBM\Firewall\config\fwrules.cfg and lists the complete service. REXX is included in the NT Resource Kit.

```

/*
services = 'c:\Program Files\IBM\Firewall\config\fwservices.cfg'
rules     = 'c:\Program Files\IBM\Firewall\config\fwrules.cfg'
out_file  = 'SERVICES.LST'

Say 'Listing' services'. Output will be in' out_file
l = time() date() 'Input files:' rules services
rc = lineout(out_file,l)

/* read rules file into memory */
numlines = lines(rules)

Do i = 1 to numlines
  If i // 500 = 0 Then Do /* say something every 500 records */
    Say Time() 'already processed' i rules 'records'
  End
  record = linein(rules)
  parse var record rul_num ']' .
  fwrule.rul_num = record
End

/* read services file */
/*numlines = number_of_lines(services)*/
numlines = lines(services)
Do i = 1 To numlines
l = ' '
  rc = lineout(out_file,l)
  If i // 500 = 0 Then Do /* say something every 500 records */
    Say Time() 'already processed' i services 'records'
  End
  record = linein(services)
  parse var record serv_num '[' comm1 '[' comm2 '[' p4 '[' p5 '[' p6 '[' p7 '[' p8 '[' .
  l = comm1 comm2
  rc = lineout(out_file,l)
/* parse p8 */
  n = Words(p8)
/* look for each rule */
  Do j = 1 to n
    rule = word(p8,j)
    parse var rule r ';' io
    parse var fwrule.r ruln '[' comm1 '[' comm2 '[' perm '[' prot '[' op1 '[' port1 '[' ,
    op2 '[' port2 '[' int '[' rout '[' dir '['
log '[' frag '[' .
    select /* make operation more readable */
      when op1 = 'eq' then opa = '='
      when op1 = 'gt' then opa = '>'
      when op1 = 'any' then opa = 'any '
      otherwise nop
    end
    select
      when op2 = 'eq' then opb = '='
      when op2 = 'gt' then opb = '>'
      when op2 = 'any' then opb = 'any '
      otherwise nop
    end
  End
/* 1st line */
if io = 'i' then flow = '==>'
else flow = '<==='
l=flow comm1 comm2
rc = lineout(out_file,l)
/* 2nd line */
l='          ' perm prot opa]]port1 opb]]port2 'Interface='int 'Routing='rout ,
'Direction='dir 'Log='substr(log,3,1) 'Frag='substr(frag,3,1)
rc = lineout(out_file,l)
End
End

```

A sample of the output follows:

```

FTP proxy in 1/2 Permit FTP inbound from non-secure network to firewall
==> Proxy FTP Control in non-secure 1/2 TCP in port 21 non-secure
    permit tcp >1023 =21 Interface=non-secure Routing=local Direction=inbound Log=n Frag=y
<== Proxy FTP Control Ack out non-secure 1/2 TCP/ACK port 21
    permit tcp/ack =21 >1023 Interface=non-secure Routing=local Direction=outbound Log=n Frag=y
<== Proxy FTP Data out non-secure 1/2 TCP port 20
    permit tcp =20 >1023 Interface=non-secure Routing=local Direction=outbound Log=n Frag=y
==> Proxy FTP Data Ack in non-secure 1/2 TCP/ACK port 20
    permit tcp/ack >1023 =20 Interface=non-secure Routing=local Direction=inbound Log=n Frag=y
==> Proxy FTP Data in non-secure 1/2 TCP port 1023+
    permit tcp >1023 >1023 Interface=non-secure Routing=local Direction=inbound Log=n Frag=y
<== Proxy FTP Data Ack out non-secure 1/2 TCP/ACK port 1023+
    permit tcp/ack >1023 >1023 Interface=non-secure Routing=local Direction=outbound Log=n Frag=y

FTP proxy in 2/2 Permit FTP inbound from firewall to secure network
==> Proxy FTP Control out secure 1/2 TCP port 21
    permit tcp >1023 =21 Interface=secure Routing=local Direction=outbound Log=n Frag=y
<== Proxy FTP Control Ack in secure 2/2 TCP/ACK port 21
    permit tcp/ack =21 >1023 Interface=secure Routing=local Direction=inbound Log=n Frag=y
<== Proxy FTP Data in secure 2/2 TCP port 20
    permit tcp =20 >1023 Interface=secure Routing=local Direction=inbound Log=n Frag=y
==> Proxy FTP Data Ack out secure 2/2 TCP/ACK port 20
    permit tcp/ack >1023 =20 Interface=secure Routing=local Direction=outbound Log=n Frag=y
==> Proxy FTP out 2/2 TCP out port 1023+ secure
    permit tcp >1023 >1023 Interface=secure Routing=local Direction=outbound Log=n Frag=y
<== Proxy FTP Ack in 2/2 TCP/ACK in port 1023+ secure
    permit tcp/ack >1023 >1023 Interface=secure Routing=local Direction=inbound Log=n Frag=y

```

The ==> and <== matches the flow arrows of the service definition.

You can download this code as detailed in “How to Access Samples from This Book” on page xiii; the name of the file is serv_lst.cmd.

Appendix H. Related Publications, Web Sites, and Mailing Lists

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

H.1 International Technical Support Organization Publications

For more information on ordering these ITSO publications, see “How to Get ITSO Redbooks” on page 629.

- *Protect and Survive: Using IBM Firewall 3.1 for AIX*, SG24-2577. This redbook is all about the IBM Firewall 3.1 for AIX, a close cousin of the product discussed in this redbook. It is also the reference for some chapters covered in this current work.
- *The Domino Defense: Security in Lotus Notes and the Internet*, SG24-4848. This redbook contains an excellent treatment on setting up firewall rules to permit Domino servers to replicate through it.
- *The Technical Side of Being an Internet Service Provider*, SG24-2133. This redbook contains relevant chapters on general Internet security.
- *TCP/IP Tutorial and Technical Overview*, GG24-3376. This is an excellent introduction to TCP/IP.
- *Java 2 Network Security*, SG24-2109. This redbook is an end-to-end blueprint for writing secure Java code, considering network and firewall implications, too.
- *A Comprehensive Guide to Virtual Private Networks, Volume I*, SG24-5201.
- *IBM Firewall for AS/400 V4R3: VPN and NAT Support*, SG24-5376.
- *IBM WebSphere Performance Pack Usage and Administration*, SG24-5233.

H.2 Redbooks on CD-ROMs

Redbooks are also available on the following CD-ROMs. Click the CD-ROMs button at <http://www.redbooks.ibm.com/> for information about all the CD-ROMs offered, updates and formats.

CD-ROM Title	Collection Kit Number
System/390 Redbooks Collection	SK2T-2177
Networking and Systems Management Redbooks Collection	SK2T-6022
Transaction Processing and Data Management Redbooks Collection	SK2T-8038
Lotus Redbooks Collection	SK2T-8039
Tivoli Redbooks Collection	SK2T-8044
AS/400 Redbooks Collection	SK2T-2849
Netfinity Hardware and Software Redbooks Collection	SK2T-8046
RS/6000 Redbooks Collection (BkMgr Format)	SK2T-8040
RS/6000 Redbooks Collection (PDF Format)	SK2T-8043
Application Development Redbooks Collection	SK2T-8037

H.3 Other Publications

These publications are relevant as further information sources:

- *Building Internet Firewalls*, D. Brent Chapman and Elizabeth D. Zwicky, (O'Reilly Associates, Inc.) SR23-7296. This book covers a great deal about firewalls, starting from firewall architectures to actual implement them. Of great value is its extensive treatment of firewall filtering rules, most of which you can apply to the IBM Firewall.
- *IBM eNetwork Firewall for Windows NT User's Guide Version 3 Release 3*. This book ships with the firewall product in electronic form. It contains basic installation and configuration instructions.
- *IBM eNetwork Firewall for Windows NT Reference Version 3 Release 3*. This book ships with the firewall product in electronic form. It contains advanced configuration options on the IBM firewall.
- *Windows NT Server 4: Professional Reference*, Karanjit S. Siyan, (New Riders Publication), ISBN 1562058053. This is a useful text that details system administration on the Windows NT Server.
- *DNS and BIND*, Third Edition, Albitz and Liu, (O'Reilly and Associates, Inc.), SR23-8771. Hands down, this is the bible for DNS and BIND.
- *Sendmail*, Second Edition, Costales and Allman, (O'Reilly and Associates, Inc.), SR23-7534. A cryptic book for a cryptic program, which is why it was indispensable for configuring Sendmail for NT when we hit our heads against limitations in SafeMail.
- *Actually Useful Internet Security Techniques*, Hughes (New Riders Publishing), ISBN 1562055089. This title holds some insight into Internet security. It's out of print, unfortunately, but can be somewhat useful if you can find an extant copy.
- *Web Security and Commerce*, Garfinkel and Spafford, (O'Reilly Associates Inc.) ISBN 1565922697. Explains risks of executable code, use of cryptography, how to set up secure Web servers.
- *Practical UNIX and Internet Security*, Garfinkel and Spafford, (O'Reilly Associates Inc.) ISBN 1565921488. Although concentrating on UNIX systems, many of the details also apply to Windows systems too.

H.4 Web Sites

Update yourself on Windows NT and Internet security issues by visiting these Web sites regularly:

- www.software.ibm.com/enetwork/firewall. This is the home page for the IBM eNetwork Firewall for Windows NT and other IBM firewall products. There is a link here to the firewall forum, where you can post a question and (hopefully) get an answer. If not, you can always pick out the e-mail addresses here and send mail directly to people who might know.
- www.cert.org. The CERT Coordination Center studies Internet security vulnerabilities, and publishes a variety of security alerts.
- www.first.org. The Forum of Incident Response and Security Teams (FIRST), brings together a variety of computer security incident response teams from government, commercial, and academic organizations.

- www.coast.com. Computer Operations, Audit, and Security Technology (COAST) is a multiple project, multiple investigator laboratory in computer security research at Purdue University.
- csrc.nist.gov. The Computer Security Resource Clearinghouse (CSRC) is designed to collect and disseminate computer security information resources to help users, systems administrators, managers, and security professionals better protect their data and systems.
- www.socks.nec.com. The SOCKS home page, maintained by NEC.
- support.microsoft.com. This is the Microsoft technical support page. The site is almost always busy and it may take some time to get in.
- www.securid.com. This is the SecurID home page, where you can pick up information on one-time passwords and security tokens.
- www.it.kth.se/~rom/ntsec.html. This is a FAQ on NT security. It has numerous links to other NT security sites.
- www.zurich.ibm.ch/Technology/Security. The IBM Zurich Security Research Group
- www.brs.ibm.com/services/brs/ers/brspwers.nsf. The IBM Emergency Response Service, a member of the IBM Global Services group.

H.5 Mailing Lists

The following are mailing lists you might want to subscribe to:

- IBM Firewalls mailing lists. The three IBM firewall products are covered in these lists. To subscribe, send a message to majordomo@computerps.com with one of the following lines in the message body:

```
subscribe ibm_firewall_aix
subscribe ibm_firewall_nt
subscribe ibm_firewall_as400
```

Leave the subject line blank.

- Comp.security.firewalls mailing lists. There are two mailing lists available here, the firewalls mailing list and the firewalls digest mailing list. This list is very active, with people posting questions and answers. A single day might see upwards of 50 messages. If you subscribe to the firewalls mailing list, this is the amount of mail you should expect to receive. The firewalls digest gives you the same content but groups the messages into chunks of 10 or so.

To subscribe to the firewalls mailing list, send a message to majordomo@greatcircle.com with the message `subscribe firewalls`. Leave the subject line blank.

To subscribe to the firewalls digest mailing list, send a message to majordomo@greatcircle.com with the message `subscribe firewalls-digest`. Leave the subject line blank.

You may also get the same information from the `comp.security.firewalls` newsgroup.

- Firewall-wizards mailing list, moderated by Marcus J. Ranum. Send a message to majordomo@nfr.net with the message `subscribe firewall-wizards`

optionally followed by your e-mail address. The list is archived as a hypernews archive on <http://www.nfr.net>.

- CERT Advisory. To subscribe to the CERT advisory, send a message to cert-advisory-request@cert.org with the subject SUBSCRIBE username@company.com. Leave the message body blank.

Also available as newsgroup comp.security.announce.

Appendix I. Special Notices

This publication is intended to help network administrators and system to plan, install, implement, and manage the IBM eNetwork Firewall for Windows NT. The information in this publication is not intended as the specification of any programming interfaces that are provided by the IBM eNetwork Firewall for Windows NT. See the PUBLICATIONS section of the IBM Programming Announcement for the IBM eNetwork Firewall for Windows NT for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The information about non-IBM ("vendor") products in this manual has been supplied by the vendor and IBM assumes no responsibility for its accuracy or completeness. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Any pointers in this publication to external Web sites are provided for convenience only and do not in any manner serve as an endorsement of these Web sites.

Any performance data contained in this document was determined in a controlled environment, and therefore, the results that may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data for their specific environment.

This document contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples contain the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

Reference to PTF numbers that have not been released through the normal distribution process does not imply general availability. The purpose of including these reference numbers is to alert IBM customers to specific information relative to the implementation of the PTF when it becomes available to each customer according to the normal IBM PTF distribution process.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

AIX	AS/400
DB2	DB2 Universal Database
eNetwork	Hummingbird
IBM	NetView
OS/2	RS/6000
SP	System/390
VisualAge	400

The following terms are trademarks of other companies:

C-bus is a trademark of Corollary, Inc.

Java and HotJava are trademarks of Sun Microsystems, Incorporated.

Microsoft, Windows, Windows NT, and the Windows 95 logo are trademarks or registered trademarks of Microsoft Corporation.

PC Direct is a trademark of Ziff Communications Company and is used by IBM Corporation under license.

Pentium, MMX, ProShare, LANDesk, and ActionMedia are trademarks or registered trademarks of Intel Corporation in the U.S. and other countries.

SET and the SET logo are trademarks owned by SET Secure Electronic Transaction LLC.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

Other company, product, and service names may be trademarks or service marks of others.

How to Get ITSO Redbooks

This section explains how both customers and IBM employees can find out about ITSO redbooks, redpieces, and CD-ROMs. A form for ordering books and CD-ROMs by fax or e-mail is also provided.

- **Redbooks Web Site** <http://www.redbooks.ibm.com/>

Search for, view, download, or order hardcopy/CD-ROMs redbooks from the redbooks Web site. Also read redpieces and download additional materials (code samples or diskette/CD-ROM images) from this redbooks site.

Redpieces are redbooks in progress; not all redbooks become redpieces and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.

- **E-mail Orders**

Send orders by e-mail including information from the redbook fax order form to:

In United States:
Outside North America:

e-mail address: usib6fpl@ibmmail.com
Contact information is in the "How to Order" section at this site:
<http://www.elink.ibmmlink.ibm.com/pbl/pbl/>

- **Telephone Orders**

United States (toll free)
Canada (toll free)
Outside North America

1-800-879-2755
1-800-IBM-4YOU
Country coordinator phone number is in the "How to Order" section at this site:
<http://www.elink.ibmmlink.ibm.com/pbl/pbl/>

- **Fax Orders**

United States (toll free)
Canada
Outside North America

1-800-445-9269
1-403-267-4455
Fax phone number is in the "How to Order" section at this site:
<http://www.elink.ibmmlink.ibm.com/pbl/pbl/>

This information was current at the time of publication, but is continually subject to change. The latest information may be found at the redbooks Web site.

IBM Intranet for Employees

IBM employees may register for information on workshops, residencies, and redbooks by accessing the IBM Intranet Web site at <http://w3.itso.ibm.com/> and clicking the ITSO Mailing List button. Look in the Materials repository for workshops, presentations, papers, and Web pages developed and written by the ITSO technical professionals; click the Additional Materials button. Employees may access MyNews at <http://w3.ibm.com/> for redbook, residency, and workshop announcements.

IBM Redbook Fax Order Form

Please send me the following:

Title	Order Number	Quantity

First name Last name

Company

Address

City Postal code Country

Telephone number Telefax number VAT number

Invoice to customer number _____

Credit card number _____

Credit card expiration date Card issued to Signature

We accept American Express, Diners, Eurocard, Master Card, and Visa. Payment by credit card not available in all countries. Signature mandatory for credit card payment.

Index

A

- ACE/Server 219, 563
- activation 159
 - rule base 159
- administration 39
- administrators 231
- AH 357
- alerts 101, 409, 420
- Archie 207
- archival process 413
- Archive Management 104, 413
- archiving 38
- Authentication 236, 297, 316, 333, 421, 424
- Authentication Header (AH) 357
- authentication profiles 18
- AUTOEXNT Service 507
- AutoSOCKS 337, 521
- Aventail AutoSOCKS 337, 521

B

- backup 497
- backup domain controller 45
- Backup Domain Controller (BDC) 302
- BAROC 585
- bastion 5
- broadcast 168
- browser 322, 335, 542

C

- cache poisoning 251
- carrier 432
- CERT Advisory CA-95:18 170
- CGI scripts 601
- chain SOCKS 336
- chaining proxy 336
- Challenge-Handshake Authentication Protocol (CHAP) 522
- Challenge-Response Authentication Method (CRAM) 18, 326, 332, 522
- clear text 312
- CNAME record 266
- commands
 - fwcfsrv 222
 - fwdfuser 296
 - fwlog 415
 - fwlogmgmt 415
 - fwmail 283
 - fwuser 299
- Common Object Request Broker Architecture (CORBA) 576

- Computer Emergency Response Team (CERT) 597
- Configuration Client 237, 240
 - port number 240
- Configuration Server 221
 - port number 221
- Connection Setup 97
- connections 33, 143, 151, 155
 - non-standard 155
 - standard 151
- CRAM 18, 326, 332, 522

D

- DB2 436, 517
- DB2/6000 436
- demilitarized zone (DMZ) 7, 78, 449
- denial of service 252
- denial-of-service 597
- DNS 19, 36, 77, 80, 81, 88, 90, 93, 188, 247, 248, 249, 252, 255, 259, 264, 266, 286, 304, 316, 499, 601
 - A record 266
 - Caching-only Servers 249
 - CNAME record 266
 - domain 247
 - forwarders 88, 252, 255
 - MX record 81, 259
 - queries 264
 - Secure DNS Server field 93
 - Secure Domain Name field 93
 - server 19, 77
 - slave 255
 - subdomain 247
 - zone transfers 249
 - zones 248
- Domain Name System (DNS) 247
- Domino 199
- dual homed firewall 169
- dual-homed gateway 6

E

- Encapsulating Security Payload (ESP) 357
- Endpoint Gateway 577
- ESP 357

F

- filter type 362
- Finger 325
- firewall 6, 29, 33, 59, 63, 69, 70
 - configuration 63
 - connections 33
 - dual-homed 6

- firewall (*continued*)
 - installation 59
 - mail server 70
 - name servers 69
 - planning 29
 - security policy 29
- firewall administrator 221
 - primary 221
- forwarders 252
- fragmentation 150
- fragmented packets 128
- FTP 80, 99, 175, 301, 306, 308, 311, 541, 549, 601
 - normal mode 80, 311
 - passive mode 80, 311
 - proxy 99
- fwcfsrv 222
- fwdfadm 232
- fwdfuser 296
- fwlog 415
- fwlogmgmt 415
- fwmail 283
- fwuser 299

G

- gateway 91, 105
- Gopher 197, 319

H

- hardening 52
- HOD 210
- Host On-Demand (HOD) 210
- HTTP 99, 195, 298, 301, 306, 316, 318, 319, 419, 498, 530, 541, 549
 - log 419
 - logging 318
 - password 298
 - proxy 99, 195, 301, 316, 419, 530
 - proxy connections 319
- HTTPS 319, 459, 541
- Hummingbird 337
- HyperText Transfer Protocol (HTTP) 192

I

- IBM Intermediate Support Driver. 12
- IBM Web Traffic Express 529
- ICMP 129, 149, 165, 202, 216, 601
- ident 201
- Integralis Technology 550
- Interface 92, 149
- Internet Scanner SAFEsuite 475
- Internet Security Association and Key Management Protocol (ISAKMP) 358

- Internet Security System 475
- Internet Service Provider (ISP) 77
- IP address spoofing 163
- IP Authentication Header (AH) 357
- IP Encapsulating Security Payload (ESP) 357
- IP forwarding 57
- IP spoofing 451
- IPSec 357
- ISP 93

J

- Java servlet 535

K

- Kerberos 522

L

- LCF gateway 578
- Lightweight Client Framework (LCF) 576
- LMHOSTS 304
- Local SPI 363
- log 412, 413, 417, 433
 - archive 413
 - audit 417
 - format 433
 - priority 412
- log control 149
- Log Viewer 417
- logging 38, 80, 244, 409, 436
- loopback addresses 167
- Lotus Notes 199, 285

M

- MAC 358
- mail 37, 89, 94, 269, 426, 498
- mailing list 427
- MAILsweeper 550
- man-in-the-middle attack 598
- Maximum Transmission Unit (MTU) 349
- message authentication code (MAC) 358
- message tag 425
- Message Threshold 102
- migration 497
- MIME 192
- MIMEsweeper 550
- mkkf 226
- modem 431, 500
- monitoring 420
- MTU 349
- Multi-Purpose Internet Mail Extensions (MIME) 192
- multicast packets 168
- MX record 259, 286

N

- name server 81
- NAT 339
 - address mappings 344
 - addresses to be excluded 343
 - connecting two networks same IP addresses 353
 - Example Configurations 351
 - FTP PORT command 341
 - ICMP 341
 - IP alias 349
 - IP Routing 348
 - Map Example Configuration 352
 - NAPT (Network Address Port Translation) 339
 - One-to-many Simulation 513
 - registered IP address 342
 - TCP and UDP Protocol Problems 341
 - timeout 350
 - Virtual Private Networks 355
- ND 503
 - Advanced usage 513
 - Advisor 506
 - DNS 509
 - Firewall Configuration 507
 - High Availability 515
 - How it Works 508
 - Installation and Configuration 504
 - Load Balancing different Services 508
 - One-to-many Simulation 513
 - Port Monitoring 509
 - SOCKS 507, 510
 - Technical Background 504
 - The Basics 503
 - The Configuration GUI 505
 - UDP 509
 - Wildcard Cluster 512
 - Wildcard Port 512
- NEC 337
- NetBIOS 303, 598
- Netscape Navigator 106, 321, 335, 542
- network adapter cards 46
- Network Address Translation 339
- Network News Transfer Protocol 189
- Network Object 95, 144
- Network Security Auditor 494
- NNTP 189
- NT registry 558

O

- OAKLEY Key Determination protocol 358
- Object Management Group (OMG) 576
- objects 143
- one-time passwords 219
- Oracle 436

P

- pager 425, 431, 500
- pager notification 426
- PASSCODE 564
- passive FTP 308
- PIN number 568
- PING 205
- ping of death 597
- Post Office Protocol (POP) 284
- Post.Office 553
- primary domain controller 45
- Primary Domain Controller (PDC) 302
- primary firewall administrator 221, 231
- Primary Name Server 249
- protocol 148, 157
- proxy 34, 35, 293, 306
 - authenticate 35
 - servers 34
 - services 34
- proxy connections 327
- proxy server 16
- PTR records 86
- Public Domain Name 270

Q

- quarantine 552
- Query Management Facility 436, 517

R

- real-time alerts 436
- RealAudio 325
- RealPlayer 325
- relay 191
- remote configuration 223
- Remote SPI 363
- reports 436
- resolvers 247
- restoration 497
- RFC 1627 78
- RFC 1918 - Address Allocation for Private Internets 339
- routing 333
- rules 143, 148
 - base design 151

S

- S-HTTP 197
- S/WAN 358
- SATAN 495
- scanning 598
- scheduler 415
- screening filter 5

- SDI authentication 574
- Secondary Name Server 249
- secure IP tunnel 357
- Secure Mail Proxy 20, 96, 269, 271, 281
- SecureWay Network Dispatcher (ND) 503
- SecurID 219, 236, 295, 302, 563
- Security Administrator Tool for Analyzing Networks (SATAN) 495
- Security Dynamics 219, 295, 563
- Security Parameters Index (SPI) 358
- security policy 29, 74, 96, 219, 263, 316
- services 145, 161, 242
- Simple Mail Transfer Protocol (SMTP) 184
- Simple Network Management Protocol (SNMP) 205
- SMTP 184, 269
 - commands 269
 - headers 270
- SMTP gateway 555
- SNMP 206
- snooping 598
- SOCKETS 558
- SOCKS 18, 35, 97, 105, 293, 302, 306, 324, 327, 331, 498
 - Challenge-Response Authentication Method (CRAM) 18
 - configuration 331
 - connections 327
 - server 166
- SOCKS Monitor 306
- SOCKS servers 522
- SOCKS traffic monitor 440
- SocksCap 337
- SOCKSified client 195
- SoftID 563
- SPI 363
- SQL 437, 517
- SSL 196, 221, 230, 459, 522
- strong authentication 564
- Structured Query Language (SQL) 436
- Synchronization 569
- System Administration 101

T

- TAP protocol 431
- TCP header 139
- teardrop attack 597
- TELEMATE.Net 436
- Telnet 170, 306, 308, 311, 601
- Telnet proxy 311
- Telocator Alphanumeric Protocol (TAP) 431
- TFTP 325
- third adapter 449
- three adapters 449
- threshold 424

- thresholds 420
- Tivoli Enterprise Console (TEC) 428
- Tivoli Management Environment (TME) 576
- TME 10 576
- TME 10 Enterprise Console (TEC) 580
- TMR Server 577
- tokens 568
- traceroute 137, 202
- transparent proxy 308
- trojan horses 599
- TTL 202
- tunnel ID 369
- Tunnel Lifetime 363

U

- UDP 202
- UDP header 141

V

- virtual private network (VPN) 357
- viruses 599
- Visualizer 517
- VPN 357
- vulnerability 480, 600

W

- WAIS 208, 319
- Web browsers 192
- Web servers 78
- WEBSweeper 550
- Whois 325
- Wide Area Information Servers 208
- Windows NT Server 44, 55
 - installation 44
 - NTFS 44
 - shares 55
- WinSock 522
- World Wide Web (WWW) 192

X

- X-Windows 325

Z

- zone transfers 249

ITSO Redbook Evaluation

Guarding the Gates Using the IBM eNetwork Firewall V3.3 for Windows NT
SG24-5209-01

Your feedback is very important to help us maintain the quality of ITSO redbooks. **Please complete this questionnaire and return it using one of the following methods:**

- Use the online evaluation form found at <http://www.redbooks.ibm.com/>
- Fax this form to: USA International Access Code + 1 914 432 8264
- Send your comments in an Internet note to redbook@us.ibm.com

Which of the following best describes you?

Customer **Business Partner** **Solution Developer** **IBM employee**
 None of the above

Please rate your overall satisfaction with this book using the scale:
(1 = very good, 2 = good, 3 = average, 4 = poor, 5 = very poor)

Overall Satisfaction _____

Please answer the following questions:

Was this redbook published in time for your needs? Yes____ No____

If no, please explain:

What other redbooks would you like to see published?

Comments/Suggestions: **(THANK YOU FOR YOUR FEEDBACK!)**

