



# Firewalls Complete

<http://kickme.to/tiger/>

Orders

Orders

Back

Forward

Comments

Comments

© 1997 The McGraw-Hill Companies, Inc. All rights reserved.  
Any use of this Beta Book is subject to the rules stated in the [Terms of Use](#).

---

# Firewalls Complete: Table of Contents

**Dedication**

**Acknowledgment**

**Preface**

*How is this book organized*

*Who should read this book?*

*About the author*

**Chapter 1**

***Internetworking Protocols and Standards: An Overview***

**Internet Protocol (IP)**

How IP Addressing Works

IP Security Risks

*IP Watcher: Hijacking the IP Protocol*

**User Datagram Protocol (UDP)**

Attacking UDP services: SATAN at easy

ISS for UNIX and Windows NT

## **Transmission Control Protocol (TCP)**

IP Addresses

*Rules*

*Classes and Masks*

## **Extending IP Addresses Through CIDR**

TCP/IP Security Risks and Countermeasure

*IP Spoofing*

*Risk of Losing Confidentiality*

*Risk of Losing Integrity*

*tcpdump - A Text-based Countermeasure*

Strobe: a Countermeasure for UNIX

IPSEC - an IETF IP Security Countermeasure

IPSO - a DoD IP Security Countermeasure

## **Routing Information Protocol (RIP)**

## **MBONE - The Multicast Backbone**

## **Internet Control Message Protocol (ICMP)**

## **Internet Group Management Protocol (IGMP)**

## **Open Shortest-Path First (OSPF)**

## **Border Gateway Protocol Version 4 (BGP-4)**

## **Address Resolution Protocol**

Reverse Address Resolution Protocol (RARP)

Security Risks of Passing IP Datagram Through Routers

## **Simple Network Management Protocol (SNMP)**

## **Watch Your ISP Connection.**

## **The Internet Protocol Next Generation or IPv6**

Address Expansion

Automatic Configuration of Network Devices

Security

Real-Time Performance

Multicasting

IPv6 Security

**Network Time Protocol (NTP)**

**Dynamic Host Configuration Protocol (DHCP)**

**Windows Sockets (WINS)**

**Domain Name System (DNS)**

Limiting DNS Information

**Firewalls Concepts**

The Flaws in Firewalls

Fun With DMZs

Authentication Issues

Trust at the Perimeter

Intranets

**From Here...**

## **Chapter 2**

### ***Basic Connectivity***

**What Happened to TTY**

What is the Baudot Code?

**UNIX to UNIX CoPy (UUCP)**

**SLIP and PPP**

**Rlogin**

**Virtual Terminal Protocol (TELNET)**

Columbia University's KERMIT: a Secure and Reliable TELNET Server

TELNET Services Security Considerations

A Systems Manager Approach to Network Security

*From Who Are You Protecting Your Network?*

*Is All the Security Efforts Worth?*

*What does Your Gut Feelings Tell You?*

*Watch for Confidentiality*

*To Err is Human!*

*Where is your Achilles Tendon?*

*The KISS Principle!*

TELNET Session Security Checklist

**Trivial File Transfer Protocol (TFTP)**

TFTP Security Considerations

**File Transfer Protocol (FTP)**

**Some of the Challenges of Using Firewalls**

**Increasing Security on IP Networks**

## **Chapter 3**

# ***Cryptography: Is it Enough?***

**Introduction**

**Symmetric Key Encryption (Private Keys)**

Data Encryption Standard (DES)

International Data Encryption Algorithm (IDEA)

CAST

Skipjack

*But is Skipjack Secure?*

RC2/RC4

**Asymmetric Key Encryption/Public Key Encryption:**

RSA

*Is RSA Algorithm Secure?*

## Digital Signature Standard (DSS)

### Message Digest Algorithms

MD2, MD4 and MD5

Secure Hash Standard/Secure Hash Algorithm (SHS/SHA)

### Certificates

Certificate Servers

*DCS: What is Under the Hood?*

The Certificate Server\*

DCS Topology\*

DCS Protocol\*

Header Section Format\*

Question Section Format\*

The DCS Record\*

### Key Management

Kerberos

*Getting to Know Kerberos Terms*

*What is in a Kerberos Session*

A Typical Kerberos Session\*

Getting a Ticket-Granting Ticket From the Kerberos Server\*

Getting Application Service Tickets for Network Services from the Kerberos Server\*

Summary Of Kerberos Authentication\*

*Cygnus' KerbNet*

Key-Exchange Algorithms (KEA)

*Diffie-Hellman Public-Key Algorithm*

### Cryptanalysis and Attacks

Ciphertext-only Attack

Known-plaintext Attack

Chosen-plaintext Attack

Adaptive-chosen-plaintext Attack

Man-in-the-middle Attack

Chosen-ciphertext Attack

Chosen-key Attack

Rubber-hose Cryptanalysis

Timing Attack

## **Cryptography Applications and Application Programming Interfaces (APIs)**

Data Privacy and Secure communications channel

*Some Data Privacy Prime and Tools*

Have a Password Policy\*

Authentication

Authenticode

NT Security Support Provider Interface (SSPI)

Microsoft Cryptographic API (CryptoAPI)

## **Cryptography and Firewalling: The Dynamic Dual**

# **Chapter 4**

## ***Firewalling Challenges: The Basic Web***

### **HTTP**

The Basic Web

What to Watch for on the HTTP Protocol

Taking Advantage of S-HTTP

Using SSL to Enhance Security

Be Careful When Caching the Web!

Plugging the Holes: a Configuration Checklist

A Security Checklist

Novell's HTTP: Better be Careful

Watch for UNIX-based Web Server Security Problems

## URI/URL

File URLs

Gopher URLs

News URLs

Partial URLs

## CGI

# Chapter 5

## ***Firewalling Challenges: The Advanced Web***

### **Extending the Web Server: Increased Risks**

ISAPI

*CGI*

*Internet Server API (ISAPI)*

*A Security Hole on IIS exploits ISAPI*

*What can you do About it?*

NSAPI

Servlets

*Servlets Applicability*

Denali

Web Database gateways

*Cold Fusion*

*Microsoft Advanced Data Connector (ADC)*

Security of E-mail Applications

Macromedia's Shockwave

*Shockwave's Security Hole*

*The Security Hole Explained*

*Countermeasures to the Shockwave Exploit*



## **Code in Web pages**

Java applets

ActiveX controls and Security Threats

*ActiveX: Silently Manipulating Security Policies*

*ActiveX Security Threat Countermeasures*

## **Chapter 6**

# ***The APIs Security Holes and Its Firewall Interactions***

## **Sockets**

BSD sockets

Windows sockets

## **Java APIs**

## **Perl modules**

## **CGI Scripts**

## **ActiveX**

ActiveX DocObjects

## **Distributed Processing**

XDR/RPC

RPC

COM/DCOM

## **Chapter 7**

# ***What is an Internet/Intranet Firewall After All?***

## **What are Firewalls After All?**

## **The Purpose of a Firewall**

The Firewall Role of Protection

## Firewalls Providing Access Control

### **The Security Role of a Firewall**

### **Promoting Privacy with a Firewall**

### **Advantages and Disadvantages of Firewalls**

Access Restrictions

Back-Door Challenges: The Modem Threat

Risk of Insider Attacks

### **Firewall Components**

Network Security Policy

*Flexibility Policy*

*Service-Access Policy*

*Firewall Design Policy*

*Information Policy*

*Dial-in and Dial-out Policy*

*Advanced Authentication*

Packet Filtering

### **Procuring a Firewall**

Needs Assessment

Buying a Firewall

Building a Firewall

Setting It Up

*Select the Hardware Required*

*Install the Necessary Software*

*Connecting and Configuring the Computer on the Network*

*Testing it*

*Adding Security Through Firewalling Software*

### **General Considerations When Installing a Firewall**

Defining a Security Policy with a Firewall Product

## **Administrating a Firewall**

Management Expertise

System Administration

## **Circuit-Level Gateways and Packet Filters**

Packet Filtering

Application Gateways

IP-Level Filtering

# **Chapter 8**

## ***How Vulnerable Are Internet Services?***

### **Protecting and Configuring Vulnerable Services**

Electronic Mail Security Threats

Simple Mail Transfer Protocol (SMTP)

*Preventing against E-mail Attacks*

*Be Careful With E-Mail Attachments*

Post Office Protocol (POP)

Multimedia Internet Mail Extensions (MIME)

### **File Transferring Issues**

File Transfer Protocol (FTP)

Trivial File Transfer Protocol (TFTP)

File Service Protocol (FSP)

UNIX-to-UNIX Copy Protocol (UUCP)

### **The Network News Transfer Protocol (NNTP)**

### **The Web and the HTTP Protocol**

Proxying HTTP

HTTP Security Holes

### **Security of Conferencing**

### **Watch This Services**

Gopher

finger

whois

talk

IRC

DNS

Network Management Station (NMS)

Simple Network Management Protocol (SNMP)

traceroute

Network File System (NFS)

## **Confidentiality and Integrity**

# **Chapter 9**

## ***Setting Up a Firewall Security Policy***

### **Assessing Your Corporate Security Risks**

Data Security

### **Understanding and Estimating the Threat**

The Virus Threat

Outside Threats

Inside Threat

### **A Word About Security Holes**

### **Setting up a Security Policy**

A Security Policy Template

# **Chapter 10**

# ***Putting It Together: Firewall design and Implementation***

## **Reviewing the Basics**

### **Selecting a Firewall**

### **Considerations About the Security Policy**

Issues to Consider About Physical Security

Issues to Consider About Access Control

Issues to Consider About Authentication

Issues to Consider About Encryption

issues to Consider About Security Auditing

Issues to Consider About Training

## **Responding to an Incident: Your Network Under Attack**

### **Dealing With an Incident**

Network Information Service as Cracking Tool

Remote Login/Shell Service as Cracking Tool

Network File System as Cracking Tool

File Transfer Protocol Service as Cracking Tool

### **To Do List in Case of an Incident**

Assessing the Situation

Cutting Off the Link

Analyze the Problem

Take Action

## **Catching an Intruder**

### **Reviewing Security**

### **Persecuting the Hacker: What the Legal System has to Say**

What The Legal System Has To Say

The Current Regulations

## **Protecting Your Corporate Site**

Preventing Break-ins at Your Site

**Final Considerations**

## **Chapter 11**

### ***Proxy Servers***

SOCKS

Tcpd, the TCP Wrapper

Setting Up and Configuring the Proxy Server

## **Chapter 12**

### ***Firewall Maintenance***

Keeping Your Firewall in Tune

Monitoring Your System

Monitoring the Unmonitored Threats

Preventive and Curative Maintenance

Preventing Security Breaches on Your Firewall

Identifying Security Holes

Recycling Your Firewall

## **Chapter 13**

### ***Firewall Toolkits And Case Studies***

The TIS Internet Firewall Toolkit

Case Studies: Implementing Firewalls

Firewalling a Big Organization: Application-Level Firewall and Package Filtering, a Hybrid System

Firewalling a Small Organization: Packet Filtering or Application-Level Firewall, a Proxy Implementation

## Firewalling in a Subnet Architecture

# Chapter 14

## ***Types of Firewalls and Products on the Market***

### **Check Points' Firewall-1 Firewall - Stateful Inspection Technology**

FireWall-1 Inspection Module

*Full State Awareness*

*Securing "Stateless" Protocols*

*The INSPECT Language*

Stateful Inspection: Under the hood

*Extensible Stateful Inspection*

*The INSPECT Engine*

*Securing Connectionless Protocols such as UDP*

*Securing Dynamically Allocated Port Connections*

*Firewall-1 Performance*

*Systems Requirements*

### **CYCON's Labyrinth Firewall - The "Labyrinth-like" System**

An Integrated Stateful Inspection

*Intelligent Connection Tracking*

*Redirecting Traffic*

*Transparent Redirection to Fault-Tolerant Systems\**

*Diverting Scanning Programs\**

*Network Address Translation*

*Load Balancing of Connections*

*Multi-Host Load Balancing\**

*Proxying - Source Address Rewriting*

*Spoofing - Destination Address Rewriting*

*IPSec - Encryption*

IPSec Filter\*

IPSec Gateway\*

Common Use\*

*Protection of Attached Networks and Hosts*

*Protection of Individual Hosts*

*Systems Requirements*

## **NetGuard's Guardian Firewall System - MAC Layer Stateful Inspection**

A Unprecedented Internet Management Tools.

*Visual Indicator of Enterprise-Wide Agent Activity:*

*Extended Gateway Information*

*Activity Monitoring Screen*

*Enhanced Activity Monitoring Screen:*

*Monitoring User's Connectivity*

*Firewall Strategy Wizard*

*WAN Adapter Support*

*Logoff Command on Authentication Client*

## **CyberGuard's CyberGuard Firewall - Hardening the OS**

The Trusted Operating System

Intuitive Remote Graphical User Interface (GUI)

Dynamic Stateful Rule Technology

Certifiable Technology

Systems Requirements

## **Raptor's Eagle Firewall - An application-level Architecture**

Enforcing Security at All Levels of the Network

Reliance on Dedicated Security Proxies

Using Raptor's Firewalls Eagle Family

*Graphical Policy Configuration*

*Consistent Management- Locally or Remote*



*The Flexibility to Allow "Transparent" Access*

*Address Redirection*

*Fine-grained control of VPN Tunnels*

*Integrated Web Blocking Capability*

HTTP Service limitations\*

Systems Requirements

## **Milkyway's SecurIT Firewall - a Factory Hardened BSDI Kernel**

A Bullet Proof Firewall

*Building a Secure Kernel*

SecurIT Firewall Kernel Modifications\*

Kernel Security Features are Certified By CSE\*

*Key Management*

Key Management and Certification Service\*

In-house Key Management\*

Manual Public Key Management\*

Private Keys\*

*Something Else You Should Know: Ubiquitous Monitoring of All Ports*

Watch for Port Numbers: The Milkyway Way\*

*Defending Against Common Attack Methods*

Buffer Overflow\*

Trojan Horses Running on the Firewall\*

Spoofing\*

Sniffing\*

Hijacking\*

Systems Requirements

## **Seattle Software's Watchguard Security Management System - Combining All Major Approaches to Firewall Design**

WatchGuard at Glance

WatchGuard Security Management System

WatchGuard's Firebox

WatchGuard's Global Console

WatchGuard Graphical Monitor

WatchGuard Reporting System

WatchGuard WebBlocker

Systems Requirements:

## **AltaVista Software's Firewall 97 - The Active Firewall**

AltaVista Firewall: Always in Motion

Services: a Matter of Security

Security: Supporting SSL

Management Features: Remote Management Through Tunneling

URL and Java Blocking

Enhanced Proxy

Powerful and Flexible Authentication

Dual-DNS Server

DMZ Support

Configuration

Hardware Requirements

## **ANS Communications's InterLock Firewall - a Dual-Homed Application Level Gateway**

ANS InterLock

ANS InterLock Service

Enhanced features in Version 4.0

InterLock's Access Controls

InterLock's Access Management

*Audit Levels*

*URL-Level Controls*

*Log Files*

*InterLock's Reports Feature*

ANS InterLock Service For Intrusion Detection

Summary of InterLock's Security Feature

**Global Technology's Gnat Box Firewall - a firewall in a floppy disk**

Getting to Know GNAT Box Firewall

*Outbound Packets from the Protected Network*

*Inbound Packets from the External Network*

*Outbound Packets from the PSN*

*How Tunnels Work in GNAT Box*

Standard Features

What is GNAT Box Firewall?

**Network-1 Software and Technology's Firewall/Plus - a High Performance Multi-Protocol Firewall**

About Firewall/Plus

Installation, Set-up and Use of FireWall/Plus

Selecting a Default Rule Base for FireWall/Plus

Performance Statistics

Additional and Advanced Filtering

Summary of Features of FireWall/Plus

*Technical Specifications*

*Special Features and General Characteristics*

Systems Requirements

**Trusted Information Systems's Gauntlet Internet - an application proxy-based Firewall**

TIS Gauntlet Internet Firewalls

A Firewall Transparent to the User

Extending Firewall Protection to Remote Offices

*Gauntlet Net Extender*

Gauntlet PC Extender

**Technologic's Interceptor Firewall - an Intuitive Firewall**

## An Overview of Technologic's Interceptor

### Interceptor's Components

*Virtual Private Networking*

*Secure Encryption for All Applications*

*Transparent Encryption for Users*

*Internet Scanner*

*The FTP Proxy*

*Telnet and Rlogin Proxy*

*HTTP Proxy*

*E-Mail Proxy*

*X11 Proxy and Generic TCP Proxy*

*The Authentication Server*

*The Domain Name Service*

*Real Audio/Real Video Proxy*

*RADAR and Utility Command Server*

*Web Caching and Java and ActiveX Blocking*

*Multiple Firewall Management*

*Systems Requirements*

## **Sun's SunScreen EFS Firewall - a Stateful Inspection Firewall**

The SunScreen Model

Secure access control.

Ease of administration.

SunScreen SPF-200 and SunScreen EFS Security Solutions

*SunScreen SPF's Features*

SunScreen SPF-200

*Features and Benefits*

SunScreen EFS

*Features and Benefits*

## System Requirements

### Solstice FireWall-1 3.0

*Solstice FireWall-1 Features*

*Comprehensive Services Support*

*Encryption Support for Data Privacy - Virtual Private Networks*

*Client Authentication*

*Anti-Spoofing and SNMP Management*

## **Secure Computing's Borderware Firewall: Combining Packet Filters and Circuit-Level Gateways**

### The BorderWare Firewall Server

*Transparency*

*Network Address Translation*

*Packet Filtering*

*Circuit-Level Gateway*

*Applications Servers*

*Audit Trails and Alarms*

*Transparent Proxies*

### BorderWare Application Services

*Mail Servers (SMTP and POP)*

*Mail Domain Name Hiding\**

*POP Mail Server\**

*Anonymous FTP Server*

*News Server*

*Web Server*

*Finger (Information) Server*

*Encryption Features*

*Automatic Backups*

### Security Features

## **Ukiah Software's NetRoad Firewall: a Multi-Level Architecture Firewall**

NetRoad FireWall for Windows NT and NetWare

*Security for Mixed Protocol (IP and IPX) Networks*

*Simple Management and NDS Integration*

*Multi-level Firewall Security and User Authentication*

NetWare and NT Firewall Support

*High Performance*

Future Evolution of the NetRoad FireWALL Platform

System Requirements

## **Secure Computing's Sidewinder Firewall: a Type Enforcement Security**

The Sidewinder Security Server

The Patented Type Enforcement Security

Remote Management

Access Controls

Extensive Event Monitoring

Advanced Filtering

*Email filtering*

*Web page filtering*

*Java applet filtering*

## **IBM's Internet Connection Secure Server Firewall: a Type Enforcement Security**

The IBM Firewall V3.1 for AIX

*Great Level of Protection*

*Greater Accessibility*

*IBM Firewall Filtering*

*IBM Firewall as an Application-Level Proxy*

*IBM Firewall as a Circuit-Level Proxy*

*Use of Encryption*

*Managing the IBM Firewall*

Main IBM Firewall Features

*Network Address Translation*

*SafeMail*

*Strong Authentication*

*Hardening*

Communicating through Virtual Private Networks

*Using the Network Security Auditor*

Administering the Firewall

*Enterprise Firewall Manager*

System requirements

## **Appendix A:**

### ***List of Firewall Resellers and Related Tools***

**AlterNet:**

**Atlantic Computing Technology Corporation**

**ARTICON Information Systems GmbH**

**Cisco Routers**

**Cohesive Systems**

**Collage Communications, Inc.**

**Conjungi Corporation**

**Cypress Systems Corporation, (Raptor reseller)**

**Data General Corp. (Gauntlet Reseller)**

**Decision-Science Applications, Inc.**

**E92 PLUS LTD**

**Enterprise System Solutions, Inc.(BorderWare reseller)**

**E.S.N - Serviço e Comércio de Informática Ltda.**

**FSA Corporation**

**IconNet**

**Igateway by Sun Consulting.**

**Ingress Consulting Group, LTD**

**INTERNET GmbH**

**Jeff Flynn & Associates**

**Media Communications eur ab, (Gauntlet Reseller)**

**Mergent International, Inc. (Gauntlet Reseller)**

**Momentum Pty Ltd**

**NetPartners (Phil Trubey), (JANUS Reseller)**

**Network Translation Services**

**OpenSystems, Inc.**

**PDC**

**PENTA**

**PRC**

**Racal-Airtech Ltd, (Eagle reseller)**

**RealTech Systems**

**Sea Change Corporation, (JANUS reseller)**

**Security Dynamics Technologies**

**Softway Pty Ltd, (Gauntlet Reseller)**

**Spanning Tree Technologies Network Security Analysis Tool**

**Stalker by Haystack Labs, Inc.**

**Stonesoft Corporation**

**TeleCommerce**

**Trident Data Systems, (SunScreen provider)**

**Tripcom Systems Inc.**

**Trusted Network Solutions (Pty) Ltd.**

**UNIXPAC AUSTRALIA**

**X + Open Systems Pty Ltd., (Internet Consultants)**

**Zeuros Limited**



# ***Firewall Tools: Public Domain and Shareware, Etc.***

**Drawbridge**

**Freestone by SOS Corporation**

**fwtk - TIS Firewall Toolkit**

**ISS**

**SOCKS**

## **Chapter 15**

### ***Glossary***

## **Bibliography & Webliography**

### ***Partial Webliography List***

[Orders](#)

[Orders](#)

[Back](#)

[Forward](#)

[Comments](#)

[Comments](#)

---

[COMPUTING MCGRAW-HILL](#) | [Beta Books](#) | [Contact Us](#) | [Order Information](#) | [Online Catalog](#)

---

HTML conversions by [Mega Space](#).

This page updated on December 05, 1997 by [Webmaster](#).

[Computing McGraw-Hill](#) is an imprint of the [McGraw-Hill Professional Book Group](#).

Copyright ©1997 [The McGraw-Hill Companies, Inc.](#) All Rights Reserved.

Any use is subject to the rules stated in the [Terms of Use](#).

|        |        |
|--------|--------|
| Orders | Orders |
|--------|--------|

|      |         |
|------|---------|
| Back | Forward |
|------|---------|

|          |          |
|----------|----------|
| Comments | Comments |
|----------|----------|

© 1997 The McGraw-Hill Companies, Inc. All rights reserved.  
Any use of this Beta Book is subject to the rules stated in the [Terms of Use](#).

---

# Preface

The Internet is an all pervasive entity in today's world of computing. To cope with the "wild" Internet, several security mechanisms were developed, among them access controls, authentication schemes and firewalls, as one of the most secure methods.

However, firewall means different things for different people. Consider the fable from India about the blind men and the elephant. Each blind man touched a different part of the elephant and came up with a totally different description. The blind man who touched the elephant's legs described it as being similar to a tree. Another blind man touched the tail and decided an elephant was like a twig. Yet another grabbed the trunk and concluded an elephant was like a snake. To some computer professionals, even to some of those in charge of Internet security, firewalls are just "walls of fire" blocking hackers outside of it. To some others, it is only a form of authentication mechanism. Some other folks consider firewalls to be synonymous with routers. Obviously, a firewall is much more than any of these individually.

The problem is only compounded by the fact that for a lot of computer and security professionals, firewalls was touched upon only fleetingly in their academic career, worse, they bumped into it at the computer room. Also, a lot of the important parts and features of firewall are recent innovations, and thus were never covered in an academic career or most of the 1995-1996 firewall books at all, which further aggravates the problem as of right now there is no one single book these professionals can turn to. Their only resource is to peruse a wide array of literature including textbooks, web pages, computer magazines, white papers, etc.

This book, the Complete Firewall Handbook, aims to become your companion book, the one you will always want to carry with you, as it does claim to be complete! I can assure you, there may be some similar books on the market, but none is complete as this one, none provides a reference guide as this one. The other titles I know are either discussing a specific technology and strategy or a product.

Although you can compare this book to those, as it also covers the firewall technologies, strategies and all the main firewall products on the market, this one goes beyond the scope of the other ones. In addition, it provides a complete reference guide of the various protocols, including the upcoming ones (IPV6, for example) and how firewalling fits into it.

In fact, this book adds a next level to your expertise by discussing all the components that makes the Internet, and any other network for that matter, unsecured: it discuss and describes in details all the protocols, standards and APIs used on internetworking, as well as the security mechanisms, from cryptograph to firewalls. Later on the book there is a "reference" section with a complete review of the

major firewall products available on the market to date, a selection of tools, applications and many firewall demos and evaluations, which were all bundled together on the CD the accompanies this book.

This book is aimed primarily to network administrators, including Web, systems, LAN and WAN Administrators. But it is also target to the new breed of professionals, the so called Internet Managers, as well as to anyone in need of a complete reference book in firewalls. As you read this book you will notice that what separates it from others is that this one is comprehensive, and gives the technical information necessary to understand, choose, install, maintain e foresee future needs involving firewalls and security at a very informal level. It has a conversational style with practical information, tips and cautions, to help the Internet, network and security administrator to cope, and "survive," their tasks and responsibilities.

As important as implementing firewalls at your site, it must be preceded of a security policy that takes in consideration services to be blocked and allowed. It should also consider implementation of authentication and encryption devices and the level of risks you are willing to undertake in order to be connected to the Internet. This book will discuss all of these topics and the issues it brings up when dealing with site security and administration. It will go over all the services, such as TELNET, FTP, Web, e-mail, news, etc.

## ***How is this book organized***

This book is organized in three parts:

**Part I**, "Introducing TCP/IP and the Need for Security: Firewalls" is a *reference* part covering all the rationale for having security at a site, the Internet threats, the security concepts and firewall fundamentals.

Chapter 1, "Internetworking Protocols and Standards: An Overview," covers all the major used on the Internet. It discusses TCP/IP, ICMP, IGMP, routing, bridging, gateways, IPv6, BGP-4, BOOTP, NTP/SNTP, DHCP, WINS, DNS and more.

Chapter 2, "Basic Connectivity," discuss the protocols and standards that enables Internet connectivity such as TTYs, UUCP, SLIP, PPP, Rlogin, Telnet, RAS and more.

Chapter 3, "Cryptography: Is it Enough?," is a natural result of what is discussed on chapter 1 and 2 in light of the insecurity of these protocols and standards. It provides and introduction on one of the most efficient techniques to enhance security on the Internet: cryptography. It provides an introduction to the subject, as well as covering symmetric encryption techniques, such as DES, IDEA, CAST, Skipjack and RC2/RC4. It also discusses asymmetric key encryption and public key encryption schemes such as RSA, PKCS, DSS and much more.

Chapter 4, "Firewalling Challenges: The Basic Web," marks the beginning of the discussion of how the insecurity and weakness of the IP technologies discussed above and the many attempts to increase its security affects services provided on the Internet. This chapter concentrates specifically on issues related to the basic Web technologies, such as HTML, URL/URI, HTTP, CGI and more.

Chapter 5, "Firewalling Challenges: The Advanced Web," digs much further into the issues discussed on chapter 4, which directly affects the Web and its level of security. This chapter discuss the concepts and

security of advanced technologies behind the Web, such as ISAPI, NSAPI, Servlets, plug-ins, ActiveX, JavaScript, Shockwave and more.

Chapter 6, "The APIs Security Holes and Its Firewall Interactions," discuss the influence of APIs on network environment connecting to the Internet and its effect due to lack of security. It covers sockets, Java APIs, Perl modules, W3C www-lib and more.

**Part II**, "Firewall Implementations and Limitations" is a more *practical* one covering all aspects involving firewall implementations considering the security limitations and advantages of plugging in security as discussed on part I in light of the multitude of protocols and standards. It discusses how to use the various types of firewall for the many different environments, and what to use where and how, etc.

Chapter 7, "What is an Internet/Intranet Firewall After All?" discusses the basic components and technology behind firewalls, extending the discussion to the advantages and disadvantages of using firewalls, security policy and types of firewalls.

Chapter 8, "How Vulnerable Are Internet Services?" lists all the major Internet services weaknesses and what can be done to minimize the risks it generates for users and corporations attached to the Internet. The chapter discusses how to protect and configure electronic mail, SMTP, POP, MIME, FTP, TFTP, FSP, UUCP, News, and much more.

Chapter 9, "Setting Up a Firewall Security Policy," peels another layer of this Internet security onion by discussing how to setup a firewall policy, what to look for and when enough security is really enough!

Chapter 10, "Putting It Together: Firewall design and Implementation," begins to put everything discussed so far into action. It discusses how to implement a firewall, from planning, chosen the right firewall according to your environment and needs, to implementing it.

Chapter 11, "Proxy Servers," is vital for the success of a firewall implementation discussed on the previous chapter. It brings security a step further by showing how proxy server can significantly enhance the level of security offered by a firewall. This chapter defines a proxy, shows how to implement it and introduces the concept of SOCKS and how to implement it with your proxy server.

Chapter 12, "Firewall Maintenance," adds naturally to the other two previous chapters. Once you setup your firewall and add a proxy server onto it you know will need to get ready for maintaining your firewall. This chapter will help you to keep your firewall in tune, monitor your systems and perform preventive and curative maintenance on your firewall.

Chapter 13, "Firewall Toolkits And Case Studies," complements this section of the book by providing you with supplementary information and study of cases on the subject.

**Part III**, "Firewall Resource Guide," expands the information contained on chapter 13 by providing an extensive resource guide on firewalls. It discusses the major firewall technologies and brands, their advantages and disadvantages, what to watch for, what to avoid, as well as what to look for in a firewall product.

Chapter 14, "Types of Firewalls," This section provides you with a technical overview of the main firewall products available on the market as of Summer of 1997. It's an extensive selection of all the major vendors and they firewall technology, so you can have a chance to evaluate each one of them before deciding which firewall best suite your needs.

Part IV, "Appendixes," provides you with specifications of the best firewalls out there, list of vendors, security companies, products and other resource utilities on firewalls, as well as a glossary of terms.

Appendix A, "List of Firewall Vendors and Products," provides you with a list of firewall vendors and their products descriptions. Most of them have a demo or evaluation copy included in the CD that accompanies this book.

Appendix B, "List of Firewall Utilities," provides a list of utilities and an overview of each one of them.

Appendix C, "Bibliography on Firewall," provides you with a list of complementary reading materials such as books, white papers, articles, etc.

Appendix D, "Webliography on Firewalls," provides you with a list of URL links of sites offering white papers, general and more technical information of firewall and proxy servers.

Appendix E, "Glossary of Terms," provide you with a comprehensive list of words and terms generally used in the firewall/Internet environment.

## ***Who should read this book?***

The professionals most likely to take advantage of this book are:

- Computer literate professionals who graduated a few or more years ago, concerned with security;
- Programmers/Analysts/Software Developers, Engineers/Test Engineers Programmers and Project Managers;
- MIS and IS&T (Information Systems and Technology) professionals;
- Professionals involved with setting up, implementing and managing Intranets and Internet;
- Webmasters;
- Entry level (in terms of computer literacy) professionals who want to understand how the Internet works rather than how to use the Internet;
- Advanced computer literate people who would use the book as a quick reference book.

## ***About the author***

Marcus Goncalves, MS in CIS, has several years of internetworking and security consulting in the IS&T arena. He lives in Southborough-MA, with his wife and kids, a bonsai tree and few tropical fishes. He's a Systems Manager for Process Software Corp., one of the leaders in Web Server technologies and TCP/IP solutions, involved with management and system analysis of Windows NT networks and Web servers.

He has thought several workshops and seminars on IS and Internet security in U.S. and internationally. He's a member of the National Computer Security Association (NCSA), the Internet Society, the Association for Information Systems (AIS) and the New York Academy of Sciences (NYAS).

He was one of the co-authors of "Web Site Administrator's Survival Guide" (Sams.Net), the author of "Protecting Your Web Site With Firewall" (PRT), the author of "Internet Privacy Kit" (Que), the co-author of "Windows NT Server 4.0: Management and Control" (PTR), and the author of "Web Security with Firewalls" (Axcel Books). Also he is a regular contributor for BackOffice Magazine, WEBster Magazine, WebWeek and Developer's Magazine.

If you're interested in his articles, check the URL <http://members.aol.com/goncalvesv/private/writer.htm>. For a complete background information, check the URL <http://members.aol.com/goncalvesv>.

If contacting the author, please send e-mail to [goncalves@process.com](mailto:goncalves@process.com) or [goncalvesv@aol.com](mailto:goncalvesv@aol.com).

Orders

Orders

Back

Forward

Comments

Comments

---

[COMPUTING MCGRAW-HILL](#) | [Beta Books](#) | [Contact Us](#) | [Order Information](#) | [Online Catalog](#)

---

HTML conversions by [Mega Space](#).

This page updated on December 05, 1997 by [Webmaster](#).

[Computing McGraw-Hill](#) is an imprint of the [McGraw-Hill Professional Book Group](#).

Copyright ©1997 [The McGraw-Hill Companies, Inc.](#) All Rights Reserved.

Any use is subject to the rules stated in the [Terms of Use](#).

|        |        |
|--------|--------|
| Orders | Orders |
|--------|--------|

|      |         |
|------|---------|
| Back | Forward |
|------|---------|

|          |          |
|----------|----------|
| Comments | Comments |
|----------|----------|

© 1997 The McGraw-Hill Companies, Inc. All rights reserved.  
Any use of this Beta Book is subject to the rules stated in the [Terms of Use](#).

---

# Chapter 1

## *Internetworking Protocols and Standards: An Overview*

It has been said that the Internet is a very dynamic place. From its efforts to emerge since earlier researching programs dated back in 1968, to its predecessor ARPANET, which much contributed for the platform of experimentation that would characterize the Internet, it all actually first came to place in 1973.

Since then, endlessly, the internetworking efforts and researching were much evolved around attending the needs for standards of the new Cyberspace communities joining the now so called the Net. Of course, you must understand that the significance of "efforts" on the Internet environment goes beyond the nature and significance of the word, it can not only be based on what the Webster would define it! Being the Internet so dynamic, so aggressive and outspoken, not only these efforts for problem resolution and standard transcends the problems and barriers coming its way, but as David Croker simply put on Lynch's and Rose's book, "Internet System Handbook" (1993), "the Internet standards process combines the components of a pragmatic engineering style with a social insistence upon wide-ranging input and review." Thus, "efforts" becomes more often the result of individual champions than of organizational planning or directives.

Unlike any other structure in the world, the Internet protocols and standards are always proposed by individual initiatives of organizations or professionals. In order to understand how new protocols emerge and eventually become standards (do they?) you will need to start getting use to the acronym RFC, or Request for Comments. This dynamics, or process, was initiated back in 1969, as a result of the dispersion of the Internet community members. These documents, as the acronym suggests, were (and are still being!) working documents, ideals, testing results, models and even complete specifications. The various members of the Internet community would read and respond, with comments, to the RFC submitted. If the idea (and grounds!) were accepted by the community, it might then become a standard.

Not much has changed in the MO (modus operandi) of the Internet community with regards to the RFCs and how they operate. However, back there in 69, there was only one network, and the community did not exceed 100 professionals. With its fast growth, the Internet began to require not only a body that would centralize and coordinate the efforts, but also "regulate" a minimum standard so that they could at

least understand and efficiently communicate among themselves.

It was around 1974 that it became clear to ARPANET that communication needed to be expanded, that not only it was necessary to accommodate multiple communications media, but also make some sense of the many domains already existent within the group. There was a need to administer this domain. It was around then that the famous TCP/IP suite begins to gain momentum, with many experiments taking place, as part of what was called Internet Experiment Notes (IEN), around 1977.

It didn't take long (1986) for the demanding discussions of the RFCs to generate a task force, composed of engineers, with the responsibility to develop standards that could effectively guide the growth of the Internet. The Internet Engineering (INENG) was created.

Today, the now called Internet Engineering Task Force (IETF) and the Internet Research Task Force (IRTF) became the two main groups responsible for a heavy load of Internet's near-term engineering requirements and long-term researching goals, both of them under the direction of the Internet Activities Board (IAB), now under a new organization called Internet Society (1992), which is the ultimately responsible for the development of Internet technologies. But if you're a veteran to the Internet, you're probably struggling with the acronym I gave for IAB, and righteously so! During its development and maturation, the IAB changed its name to Internet Architecture Board (from Activities to Architecture), as IAB did not really had much to do with the operating part of the Internet development.

In terms of relying of RFCs as a standard, the first one to be considered so was the RFC 733. If you have an idea for a standard, or a new technology that can benefit the Internet, you will need to submit it as an RFC to the community. As a member of the IAB, the RFC Editor is the one that "moderates" the release of RFCs. As any official document, the RFCs have a style and format.

### **Tip:**

If you want to get the RFC style guide, you should refer to RFC 1111. For more information about submitting an RFC, send an e-mail message to [rfc-editor@isi.edu](mailto:rfc-editor@isi.edu). For a list of RFCs, retrieve the file `rfc/rfc-index.txt`.

### **Note:**

For more detailed information about the IAB, the IETF and the IRTF, I suggest you to get Lynch and Rose's book, "Internet System Handbook," as it's not the scope of this book to discuss the specifics of it.

It's not the scope of this book to discuss every protocol used on the Internet. I have for that at least couple reasons:

1. These protocols are too many and in constant change (and will continue to change), so this book wouldn't be of service to you, and
2. Our goal here is to concentrate on the security flaws specific to each of these protocols. By assessing their security issues not only you will be able to make a more informed decision when



choosing a protocol but also understand why all these efforts and fuzz on security alternatives such as cryptography, firewalls and proxy servers becomes necessary.

Therefore, this chapter focus on discussing the major Internet protocols, their characteristics, weaknesses and strength, and how they affects your connectivity and data exchange on the Internet. Table 1.1 provides you a list of the major protocols in used on the Internet.

**Table 1.1**

**RFCs sent to IETF on IP Support**

| <b>RFC #</b> | <b>Description of the Document</b>                                                      |
|--------------|-----------------------------------------------------------------------------------------|
| 768          | User Datagram Protocol (UDP)                                                            |
| 783          | Trivial File Transfer Protocol (TFTP)                                                   |
| 791          | Internet Protocol (IP)                                                                  |
| 792          | Internet Control Message Protocol (ICMP)                                                |
| 793/1323     | Transmission Control Protocol (TCP)                                                     |
| 826          | Address Resolution Protocol (ARP)                                                       |
| 854          | Virtual Terminal Protocol (Telnet)                                                      |
| 877/1356     | IP over X.25 Networks                                                                   |
| 903          | Reverse Address Resolution Protocol (RARP)                                              |
| 904          | Exterior Gateway Protocol (EGP) Version 2                                               |
| 950          | Internet Subnetting Procedures                                                          |
| 951          | Bootstrap Protocol (BootP)                                                              |
| 1001         | Protocol Standard for a NetBIOS Service on a TCP/UDP Transport: Concept and Methods     |
| 1002         | Protocol Standard for a NetBIOS Service on a TCP/UDP Transport: Detailed Specifications |
| 1009         | Internet Gateway Requirements                                                           |

|      |                                                                   |
|------|-------------------------------------------------------------------|
| 1042 | IP over IEEE 802 Networks                                         |
| 1058 | Routing Information Protocol (RIP)                                |
| 1063 | Maximum Transmission Unit Discovery Option                        |
| 1075 | Distance Vector Multicast Routing Protocol (DVMRP)                |
| 1084 | BootP Vendor Extensions                                           |
| 1108 | Revised Internet Protocol Security Option (RIPSO)                 |
| 1112 | Internet Group Management Protocol                                |
| 1155 | Structure and Identification of Management Information            |
| 1156 | Internet Management Information Base                              |
| 1157 | Simple Network Management Protocol (SNMP)                         |
| 1188 | IP over FDDI                                                      |
| 1247 | Open Shortest Path First (OSPF) Version 2                         |
| 1256 | Router Discovery                                                  |
| 1267 | Border Gateway Protocol (BGP) Version 3                           |
| 1519 | Classless Inter-Domain Routing (CIDR)                             |
| 1532 | Clarification's and Extension to BootP for the Bootstrap Protocol |
| 1533 | DHCP Options and BootP Vendor Extensions                          |
| 1542 | Clarification's and Extension to BootP for DHCP                   |
| 1654 | BGP Version 4                                                     |

# Internet Protocol (IP)

The Internet Protocol (IP) is considered the network protocol mostly used by corporations, governments, and the Internet. It supports many personal, technical, and business applications, from e-mail and data processing to image and sound transferring.

IP features a connectionless datagram (a packet) delivery protocol that performs addressing, routing, and control functions for transmitting and receiving datagrams over a network. Each datagram includes its source and destination addresses, control information, and any actual data passed from or to the host layer. This IP datagram is the unit of transfer of a network (Internet included!). Being a connectionless protocol, IP does not require a predefined path associated with a logical network connection. As packets are received by the router, IP addressing information is used to determine the best route that a packet can take to reach its final destination. Thus, even though IP does not have any control of data path usage, it is able to re-route a datagram if a resource becomes unavailable.

## *How IP Addressing Works*

There is a mechanism within IP that enables hosts and gateways to route datagrams across the network. This IP routing is based on the destination address of each datagram. When IP receives a datagram, it checks a header, which is present in every datagram, searching for the destination network number and a routing table. All IP datagrams begin with this packet header, illustrated on figure 1.1., which lists:

- The version of IP protocol used to create the datagram,
- The header length,
- The type of service required for the datagram,
- The length of the datagram,
- The datagram's identification number,
- The fragmentation control information,
- The maximum number of hops the datagram can be transported over the Internet/Intranet,
- The protocol format of the data field,
- The source and destination addresses, and even
- IP options.

All the datagrams with local addresses are delivered directly by the IP, and the external ones are forwarded to their next destination based on the routing table information.

IP also monitors the size of a datagram it receives from the host layer. If the datagram size exceeds the maximum length the physical network is capable of sending, then IP will break up the datagram into smaller fragments according to the capacity or the underlying network hardware. These datagrams are then reassembled at its destination before it is finally delivered.

IP connections are controlled by IP addresses. Every IP address is a unique network address that identifies a node on the network, which includes protected (LANs, WANs and Intranets) as well as unprotected ones such as the Internet. IP addresses are used to route packets across the network just like

the U.S. Postal Office uses ZIP codes to route letters and parcels throughout the country (internal network, which it has more control) and internationally (external network, which it has minimum control, if any!).

In a protected network environment such as a LAN, a node can be a PC using a simple LAN Workplace for DOS (LWPD), in which case the IP address is set by modifying a configuration file during installation of the LWPD software.

The Internet Protocol is the foundation of the Transmission Control Protocol/Internet Protocol (TCP/IP), a suite of protocols created especially to connect dissimilar computer systems, which is discussed in more details later on this chapter.

## ***IP Security Risks***

If there were no security risk concerns about connectivity on the Internet, there would not be a need for firewalls and other defense mechanisms either, and I probably would be already in God's ministry somewhere in the world, rather than writing a book about it. Thus, the solutions to the security concerns of IP-based protocols are widely available in both commercial and freely available utilities, but as you will realize throughout this book, most of the times a system requires administrative effort to properly keep the hackers at bay.

Of course, as computer security becomes more of a public matter, it is nearly impossible to list all of the tools and utilities available to address IP-based protocols security concerns. Throughout this book you are introduced to many mechanisms, hardware technologies and application software to help you audit the security of your network, but for now, let's concentrate on the security weaknesses of the protocols used for connections over the Internet by identifying the flaws and possible workarounds and solutions.

### **IP Watcher: Hijacking the IP Protocol**

There is a commercial product called IP Watcher, as showing on figure 1.2, that is capable of hijacking IP connections by watching Internet sessions and terminating or taking control over them whenever and administrator (or a hacker!) needs it. A quick click on the list of open connections shows the current conversation and everything that is being typed. Another click and the user is permanently put on hold while IP Watcher takes over the conversation. Needless to say, the evil use for this software are nearly limitless.

But IP Watcher is not the only product you should be concerned about when thinking of the security of your IP connections. There are many other crude tools for hijacking connections among the hacker community. To me, the beauty of IP Watcher (and threat!) is that it makes it point-and-click easy.

The symptoms of being "IP Watched" are minimum and misleading, but yet noticeable. If you are experiencing extreme delays on the delivery of datagrams to the point of your server eventually timing-out can be a strong indication that your IP connections are being hijacked. Also, if you are a network administrator, familiar with sniffers and have on handy, watch what is usually referred to as an "ACK storm." When someone hijacks an IP connection it generates a storming attempt on the server (or workstation!) trying to reconnect the session, which causes a heavy spamming on the network.

There are many other advanced tools out there to intercept an IP connection, but they are not easily available. Some even have the ability to insert data into a connection while you are reading your e-mail, for example, whereas suddenly all your personal files could start being transmitted across the wires to a remote site. The only sign you would notice would be a small delay on the delivery of the packets, but you wouldn't notice it while reading your e-mail or watching a disguising porno video on the Web! But don't go "bazuka" about it! Hijacking an IP connection is not as easy as it sounds when reading this paragraphs! It requires the attacker to be directly in the stream of the connection, which in most cases forces the him/her to be at your site.

### Tip:

If you want to learn more about similar tools for monitoring or hijacking IP connections on the Internet and protected networks, check the following sites below:

- **<http://cws.iworld.com>** - This site provides several 16 and 32-bits Windows (NT and Windows 95) Internet tools.
- **<http://www.uhsq.uh.edu>** - You will find several UNIX security tools in this site, with short and comprehensive descriptions for every tool.
- **<ftp://ftp.bellcore.com/pub/nmh>, <ftp://primal.iems.nwu.edu/pub/skey>** - This site maintains the core S/Key software.
- **<ftp://ftp.funet.fi>** - Here you will find general security/cracking utilities such as npasswd, passwd+, traceroute (as showing on figure 1.3), whois, tcpdump, SATAN, and Crack. For faster searching of utilities, once in the site use 'quote site find <find>', where <find> is the phrase to look for on the file-system. Using a web client, use '<http://ftp.funet.fi/search:<find>>'.

One more thing. Be careful with the information you provide the InterNIC! If you need a site on the Internet you must apply for a domain name with InterNIC. When you do that, you must provide information about the administrative and technical contact at your organization, with their phone numbers, e-mail addresses, and a physical address for the site. Although this is a good safe measure, if someone issues the UNIX command 'whois <domainname>,' as showing on figure 1.4, the utility will list all of that information you provided InterNIC with.

Not that you should refuse to provide the information to InterNIC. This is a requirement and also used for your protection as well, but when completing this information keep in mind that hackers often use it to find out basic information about a site. Therefore, be conservative, be wise. For the contact names, for example, use an abbreviation or a nick name. Consulting the information at InterNIC is usually the starting point for many attacks to your network.

During the spring of 1997, while coordinating a conversion from MS Mail to MS Exchange my mailer went South (mea culpa!) and few listservers where spammed as a result. Within hours one of our systems manager was getting a complaining phone call, at his home phone number, and the complainer knew exactly who to ask for! By using 'whois' the sysop of the spammed listserver was able to identify the name and address of the company I work for. Since it was a weekend, he could not talk to anyone about the problem, but with the systems manager's name and the city location of our company, the sysop only

had to do a quick search at query engines such as Four11 (<http://www.four11.com>) to learn the home address and phone number of our systems manager!

## User Datagram Protocol (UDP)

User Datagram Protocol (UDP), as documented on RFC 768, provides an unreliable, connectionless datagram transport service for IP. Therefore, this protocol is usually used for transaction-oriented utilities such as the IP standard Simple Network Management Protocol (SNMP) and Trivial File Transfer Protocol (TFTP).

Like TCP, which is discussed in the next section, UDP works with IP to transport messages to a destination and provides protocol ports to distinguish between software applications executing on a single host. However, UDP avoids the overhead of reliable data transfer mechanism by not protecting against datagram loss or duplication, unlike TCP. Therefore, if your data transferring requires reliability of its delivery you should definitely avoid UDP and use TCP. Figure 1.5 shows the format of an UDP header.

## *Attacking UDP services: SATAN at easy*

SATAN, a popular tool for auditing networks, is freely available for UNIX systems. SATAN is an Internet-based tool that has the ability of scanning open UDP services (as well as TCP) running on systems and provides a low level of vulnerability checking on the services it finds.

Although most of the vulnerabilities it detects have been corrected in recent operating systems, SATAN is still widely used for checking (or if you're a hacker, learning!) the configuration of systems. The tool is easy to use, but it is a bit slow and can be inaccurate when dealing with unstable networks.

SATAN runs under X-windows on UNIX and a version can be found for most flavors, with a patch required for Linux. Be careful when using the tool on its heaviest scan setting, as it usually ends up setting off alarms for vulnerabilities that have been out of date for years.

## *ISS for UNIX and Windows NT*

The Internet Security System (ISS), as showing on figure 1.6, is a scanning suite of products are commercially available for scanning Web servers, firewalls, and internal hosts. The suite includes a great deal of the latest Internet attacks and system vulnerabilities for probing UDP services (as well as TCP). It can be configured for periodic scanning and has several options for report generation, including export to a database.

The level of the attacks included and the highly customizable nature of ISS far surpass SATAN as an auditing tool. Figure 1.7 shows a screenshot of ISS Web site, where an evaluation copy of the product can be downloaded. In its evaluation version, the program will only scan the machine its installed on, but a cryptographic key can be purchased from ISS that will allow a further machines to be scanned.

Several large companies use the product internally to check the configuration of their systems and to certify firewalls for sale or for use within their organization. The product is currently available for

several flavors of UNIX and Windows NT and is currently priced based on the size of a site's network.

# Transmission Control Protocol (TCP)

Transmission Control Protocol (TCP) provides a reliable, connection-oriented, transport layer service for IP. Due to its high capability of providing interoperability to dissimilar computer systems and networks, TCP/IP has rapidly extended its reach beyond the academic and technical community into the commercial market.

Using a handshaking scheme, this protocol provides the mechanism for establishing, maintaining, and terminating logical connections between hosts. Additionally, TCP provides protocol ports to distinguish multiple programs executing on a single device by including the destination and source port number with each message. TCP also provides reliable transmission of byte streams, data flow definitions, data acknowledgments, data retransmission, and multiplexing multiple connections through a single network connection.

Of course, this section is not aimed to provide you with all the ins and outs of TCP/IP networking. For that I suggest you to read the RFC 1323 (Van Jacobson TCP), and other bibliographic references listed at the end of this book. However, in order for you to understand the security weaknesses of this protocol, it is important for us to review the general TCP/IP concepts and terminology as well as the extensive flexibility and capability that not only contributes to its widely acceptance as an Internet protocol but also its security flaws.

## IP Addresses

All the IP-based networks (Internet and LANs and WANs) use a consistent, global addressing scheme. Each host, or server, must have a unique IP address. Some of the main characteristics of this address scheme are:

- Addresses cannot be duplicated, so they won't conflict with other networks on the Internet,
- IP addressing allows an unlimited number of hosts or networks to connect to the Internet and other networks,
- IP addresses allow networks using different hardware addressing schemes to become part of dissimilar networks

## Rules

IP addresses are composed of four one-byte fields of binary values separated by a decimal point. For example,

1.3.0.2      192.89.5.2      142.44.72.8

An IP address must conform to the following rules:

- The address consists of 32 bits divided into four fields of one byte (eight bits) each.

- It has two parts: a network number and a host or machine number.
- All hosts on the same network must have the same network number.
- No two hosts on the same network can have the same host number.
- No two networks can have the same network number if they are connected in any way.

But to remember all this numbers can be hard and confusing. Therefore, in IP addressing, a series of alpha characters, known as the host name address, are also associated with each IP address. Another advantage for using the host name address is that IP addresses can change as the network grows. The full host name is composed of the host name and the domain name.

For example, the full host name for Process Software's Web server CHEETAH.PROCESS.COM is composed of the host name CHEETAH and the domain PROCESS.COM, or the IP address 198.115.138.3, as shown on figure 1.8.

**Tip:**

You can always find the IP address of a host or node on the Internet by using the PING command, as shown on figure 1.9.

The host names will be determined usually by LAN Administrator, as he/she adds a new node to the network and enters with its address on the DNS (Domain Name Service) database.

**Tip:**

Never assign a host name to a specific user or location of a computer as these characteristics tend to change frequently. Also, keep your host names short, easy to spell, free of numbers and punctuation.

## Classes and Masks

There are three primary IP categories or address classes. An IP address class is determined by the number of networks in proportion to the number of hosts at an internet site. Thus, a large network like the Internet can use all three internet address classes. The address classes are as follows:

- **Class A** — Uses the first byte for the network number and the remaining three bytes for the host number. The first byte ranges in decimal value from 1 to 127, which allows up to 128 networks and up to 16,777,216 hosts per network.
- **Class B** — Uses the first two bytes for the network number and the last two bytes for the host number. The first byte ranges in decimal value from 128 to 191, which allows up to 16,384 networks and up to 65,536 hosts per network.
- **Class C** — Uses the first three bytes for the network number and the last byte for the host number. The first byte ranges in decimal value from 192 to 223, which allows up to 2,097,152 networks, and less than 256 hosts per network.



The address class determines the network mask of the address. Hosts and gateways use the network mask to route internet packets by:

1. Extracting the network number of an internet address.
2. Comparing the network number with their own routing information to determine if the packet is bound for a local address

The network mask is a 32-bit internet address where the bits in the network number are all set to one and the bits in the host number are all set to zero.

Table 1.2 lists the decimal value of each address class with its corresponding network mask. The first byte of the address determines the address class. Figure 1.9 shows the decimal notation of internet addresses for address classes A, B, and C.

**Table 1.2 - Internet Address Classes**

| Address Class Mask | First Byte   | Network Mask  |
|--------------------|--------------|---------------|
| A                  | 1. to 127.   | 255.0.0.0     |
| B                  | 128. to 191. | 255.255.0.0   |
| C                  | 192. to 233. | 255.255.255.0 |
| D                  | 224. to 239  | None          |

**Note:**

Class D addresses are used for multicasting. Values 240 to 255 are reserved for Class E, which are experimental and not currently in use.

## Extending IP Addresses Through CIDR

In 1992, the Internet Engineering Steering Group (IESG) determined that Class B addresses assigned to hosts were quickly becoming exhausted and inefficiently used. This problem demanded a quick solution, which resulted in the development of an Internet standard track protocol, called the Classless Inter-Domain Routing (CIDR) protocol (RFCs 1517-19).

CIDR replaces address classes with address prefixes, the network mask must accompany the address. This strategy conserves address space and slows the increasing growth of routing tables. For example, CIDR can aggregate an IP address, which is called a supernet address, in the form of 192.62.0.0/16, where 192.62.0.0 represents the address prefix, and 16 is the prefix length in bits. Such an address represents destinations from 192.62.0.0 to 192.62.255.255. CIDR is supported by OSPF and BGP-4, which are discussed in more details later on this chapter.

# ***TCP/IP Security Risks and Countermeasure***

As you probably already figured out, security is not a strong point of TCP/IP, at least with the current version IPv4 (Internet Protocol version 4). Although it is not possible to have a 100% secure network, the information within these networks must be accessible to be useful. Thus, it's the balancing of accessibility and security that will define the tradeoffs management must consider and in turn decides on a security policy that supports the risks and needs of the company in accessing the Internet.

Many of the global Internet's security vulnerabilities are inherent in the original protocol design. There are no security features built into IPv4 itself, and the few security features that do exist in other TCP/IP protocols are weak. A sound internetworking security involves and requires a careful planning and development of a security policy so that unauthorized access can be prevented and difficult to achieve, as well as easy to detect.

There have been many devices developed to add security to TCP/IP networks. Also internal policies normally allow users in the protected network to free communicate with all other users on this same network, but access to remote systems and external networks (Internet) are usually controlled through different levels of access security.

Access strategies can range from quite simple to complex. A password could be required to gain access to a system, or complex encryption schemes might be required instead, as discussed in chapter 3, "Cryptography: Is it Enough?"

The most common adopted Internet security mechanism is the so called firewall, which is briefly discussed at the end of this section and extensively covered from chapter 4 on, where various environment and products are covered. But most security features that do exist in the TCP/IP protocols are based on authentication mechanisms. Unfortunately the form of authentication most often used is based on insecure IP addresses or domain names, which are very easy to be broken.

## **IP Spoofing**

A common method of attack, called IP spoofing involves imitating the IP address of a "trusted" host or router in order to gain access to protected information resources. One avenue for a spoofing attack is to exploit a feature in IPv4 known as source routing, which allows the originator of a datagram to specify certain, or even all intermediate routers that the datagram must pass through on its way to the destination address. The destination router must send reply datagrams back through the same intermediate routers. By carefully constructing the source route, an attacker can imitate any combination of hosts or routers in the network, thus defeating an address-based or domain-name-based authentication scheme.

Therefore, you can say that you have been "spoofed" when someone, by-passing source routing, trespass it by creating packets with spoofed IP addresses. Yeah, but what is this "IP spoofing" anyway?

Basically, spoofing is a technique actually used to reduce network overhead, especially in wide area networks (WAN). By spoofing you can reduce the amount of bandwidth necessary by having devices, such as bridges and routers, answer for the remote devices. This technique fools (spoofs) the LAN device into thinking the remote LAN is still connected, even though it is not. However, hackers use this same technique as a form of attack on your site.

Figure 1.10 explains how spoofing works. Hackers can use of IP spoofing to gain root access, by creating packets with spoofed source IP addresses. This tricks applications that use authentication based on IP addresses and leads to unauthorized user and very possibly root access on the targeted system. Spoofing can be successful even through firewalls if they are not configured to filter incoming packets whose source address are in the local domain.

You should also be aware of routers to external networks that are supporting internal interfaces. If you have routers with two interfaces supporting subnets in your internal network, be on alert, as they are also vulnerable to IP spoofing.

**Tip:**

For additional information on IP spoofing, please check Robert Morris paper "A Weakness in the 4.2BSD UNIX TCP/IP Software," at URL [ftp://research.att.com:/dist/internet\\_security/117.ps.Z](ftp://research.att.com:/dist/internet_security/117.ps.Z)

When spoofing an IP to crack into a protected network hackers (or crackers, for that matter!) are able to bypass one-time passwords and authentication schemes by waiting until a legitimate user connects and login to a remote site. Once the user's authentication is complete, the hacker seizes the connection, which will compromise the security of the site thereafter. This is more common among the SunOS 4.1.x systems, but it is also possible in other systems.

You can detect an IP spoofing by monitoring the packets. You can use netlog, or similar network-monitoring software to look for packet on the external interface that has both addresses, the source and destination, in your local domain. If you find one, this means that someone is tampering onto your system.

**Tip:**

Netlog can be downloaded through anonymous FTP from URL:  
<ftp://net.tamu.edu:/pub/security/TAMU/netlog-1.2.tar.gz>

Another way for you to detect IP spoofing is by comparing the process accounting logs between systems on your internal network. If there has been an IP spoofing, you might be able to see a log entry showing a remote access on the target machine without any corresponding entry for initiating that remote access.

As mentioned before, the best way to prevent and protect your site from IP spoofing is by installing a filtering router that restricts the input to your external interface by not allowing a packet through if it has a source address from your internal network. Following CERT's recommendations, you should also filter outgoing packets that have a source address different from your internal network in order to prevent a source IP spoofing attack originating from your site, as shown on figure 1.11, but much more will be discussed about it on the chapters to come.

**Caution:**

If you believe that your system has been spoofed, you should contact the CERT Coordination Center or your representative in Forum of Incident Response and Security Teams (FIRST).

CERT staff strongly advise that e-mail be encrypted. The CERT Coordination Center can support a shared DES key, PGP (public key available via anonymous FTP on [info.cert.org](http://info.cert.org)), or PEM (contact CERT staff for details).

Internet E-mail: [cert@cert.org](mailto:cert@cert.org) or Telephone: +1 412-268-7090 (24-hour hotline)

## Risk of Losing Confidentiality

The IP layer does provide some sort of support for confidentiality. One of the most common used one is the Network Encryption System (NES), by Motorola, which provides datagram encryption. The problem is that NES encryption totally seals off the protected network from the rest of the Internet.

Although NES is used to some extent among the military services to provide IP network security for the different levels of classified data, this strategy is near to unacceptable for corporate use. Besides, NES have a very elaborated configuration scheme, low bandwidth, and does not support IP Multicast.

## Risk of Losing Integrity

The TCP/IP protocol also has some schemes to protect data integrity at the transport layer by performing error detection using checksums. But again, in the sophisticated Internet environment of today, much different from the early 80's, simple checksums are inadequate. Thus, integrity assurance is being obtained through the use of electronically signatures, which as a matter of fact, are not currently part of IPv4.

Nevertheless, there are prototype integrity mechanisms among the security features for IPv4, which also are being incorporated into IPv6, that have been produced by the IETF IPSEC Working Group.

## tcpdump - A Text-based Countermeasure

Sometimes network problems require a sniffer to find out which packets are hitting a system. The program 'tcpdump,' as showing at works on figure 1.12 produces a very unintelligible output that usually requires a good networking manual to decode. But for those that brave the output, it can help solve network problems, especially if a source or destination address is already known. As for just perusing the information on the wire, it can be less than hospitable.

The sniffer 'tcpdump' can be found on most UNIX security archives and requires the 'libpcap' distribution to compile. It compiles on a wide variety of systems, but for certain machines, such as Suns, special modifications have to be made to capture information sent from the machine its installed on.

## ***Strobe: a Countermeasure for UNIX***

The utility ‘strobe,’ as showing on figure 1.13, is available from most UNIX repositories and is used to check just TCP services on a system. Sometimes, this is sufficient to check the configuration of systems. It works only as a text tool for UNIX and misses UDP, which is primarily DNS and a small selection of other services. The utility prints line by line what is available on a system and is useful for systems that enjoy scripting management tools.

Strobe is easy to run and will compile on most flavors of UNIX. It can be obtained from most popular UNIX security archives.

## ***IPSEC - an IETF IP Security Countermeasure***

The Internet Protocol Security Architecture (IPSEC) is a result of the works of the Security Working Group of the IETF, which realized that IP needed stronger security than it had. In 1995 IPSEC was proposed as an option to be implemented with IPv4 and as an extension header in IPv6 (the IPv6 suite discussed later on this chapter).

IPSEC supports authentication, integrity and confidentiality at the datagram level. Authentication and integrity are provided by appending an authentication header option to the datagram, which in turn makes use of public-key cryptography methods and openly available algorithms. Thus, confidentiality is also provided by the IP encapsulating security payload (ESP). ESP encrypts the datagram payload and header and attaches another cleartext header to the encrypted datagram, which can also be used to set up private virtual networks within the Internet.

## ***IPSO - a DoD IP Security Countermeasure***

The IP Security Option (IPSO) was proposed by the Department of Defense (DoD) in 1991 as a set of security features for the IPv4 suite. IPSO consists of two protocols for use with the Internet protocol:

- **The DoD Basic Security Option (BSO)** - The BSO protocol defines the content of the access control sensitivity labels to be attached to IP datagrams coming into and leaving the system,
- **The DoD Extended Security Option (ESO)** - The ESO protocol describes the requirements and mechanism to increase the number of hierarchical security classifications and protection authorities.

The scheme consists in labeling datagrams with their level of sensitivity in much the same way that government agencies label and control classified documents (Top Secret, Secret, Confidential, and Unclassified), but without any encryption scheme. Maybe because of it, IPSO never made it as an Internet Standard and no implementations exists.

# Routing Information Protocol (RIP)

Routing Information Protocol (RIP) is a distance-vector, interior gateway protocol (IGP) used by routers to exchange routing information, as shown on figure 1.14. Through RIP, endstations and routers are provided with the information required to dynamically choose the best paths to different networks.

RIP uses the total number of hops between a source and destination network as the cost variable in making best path routing decisions. The network path providing the fewest number of hops between the source and destination network is considered the path with the lowest overall cost.

The maximum allowable number of hops a packet can traverse in an IP network implementing RIP is 15 hops. By specifying a maximum number of hops, RIP avoids routing loops. A datagram is routed through the internetwork via an algorithm that uses a routing table in each router. A router's routing table contains information on all known networks in the autonomous system, the total number of hops to a destination network, and the address of the "next hop" router in the direction of the destination network.

In a RIP network, each router broadcasts its entire RIP table to its neighboring router every 30 seconds. When a router receives a neighbor's RIP table, it uses the information provided to update its own routing table and then sends the updated table to its neighbors.

This procedure is repeated until all router's have a consistent view of the network topology. Once this occurs, the network has achieved convergence, as shown on figure 1.15.

## ***The Multicast Backbone***

The Multicast backbone (MBONE) is a very important component when transmitting audio and video over the Internet. It was originated from the first two IETF "audiocast" experiments with live audio and video multicast from the IETF meeting site to destinations around the world. The whole concept is to construct a semi-permanent IP multicast testbed to carry the IETF transmissions and support continued experimentation between meetings, which by the way, is a cooperative, volunteer effort.

As a virtual network, MBONE is layered on top of portions of the physical Internet to support routing of IP multicast packets. Topologically, the network is composed of islands linked by virtual point-to-point links called "tunnels." These tunnels usually lead to workstation machines with operating systems supporting IP multicast and running the "mrouted" multicast routing daemon.

You might want to enroll your Web site in this effort. It will allow your Web users to participate in IETF audiocasts and other experiments in packet audio/video, as well as help you and your users to gain experience with IP multicasting for a relatively low cost.

To join the MBONE is not complicated. You will need to provide one more IP multicast routers to connect with tunnels to your users and other participants. This multicast router will usually be separate from your main production router, as most of these routers do not support multicast. Also, you will need to have workstations running the mrouted program.

You should allocate a dedicated workstations to the multicast routing function. This will prevent from other activities interfering with the multicast transmission, and you will not have to worry about installing kernel patches or new code releases on short notice that could affect that functionality of other applications. Figure 1.16 is a typical layout of an MBONE configuration:

**Figure 1.16****MBONE Configuration screen**

The configuration shown on figure 5.20 allows mrouted machine to connect with tunnels to other regional networks over the external DMZ and the physical backbone network, and connect with tunnels to the lower-level mrouted machines over the internal DMZ, thereby splitting the load of the replicated packets.

The only problem in promoting MBONE is that the most convenient platform for it is a Sun SPARCstation. You can use a VAX or MicroVAX, or even a DecStation 3100 or 5000, running Ultrix 3.1c, 4.1, 4.2a. But our typical Web server OS won't do it. In this case, you must rely on Internet Service Providers (ISP).

**Note:**

The following is a partial list of ISP who are participating in the MBONE:

AlterNet - ops@uunet.uu.net

CERFnet - mbone@cerf.net

CICNet - mbone@cic.net

CONCERT - mbone@concert.net

Cornell - swb@nr-tech.cit.cornell.edu

JANET - mbone-admin@noc.ulcc.ac.uk

JvNCnet - multicast@jvnc.net

Los Nettos - prue@isi.edu

NCAR - mbone@ncar.ucar.edu

NCSAnet - mbone@cic.net

NEARnet - nearnet-eng@nic.near.net

OARnet - oarnet-mbone@oar.net

PSCnet - pscnet-admin@psc.edu

PSInet - mbone@nisc.psi.net

SESQUINET - sesqui-tech@sesqui.net

SDSCnet - mbone@sdsc.edu

SURAnet - multicast@sura.net

UNINETT - mbone-no@uninett.no

One of the limitations of Mbone is with regards to audio capabilities, which is still troublesome, specially with Windows NT system, as it requires you to download the entire audio program before it can be heard. Fortunately, there are now systems available which avoid this problem by playing the audio as it is downloaded. The following is a list of some of them that I have tested with Windows 95 and Windows NT 3.51 and 4.0 Beta 2:

- **RealAudio** - Developed by Progressive Networks. You can download an evaluation copy from their URL at: <http://www.realaudio.com>. This player communicates with a specialized RealAudio server in order to play back audio as it is downloaded, which eliminates the delays during download, especially with slow modems. It also supports a variety of quality levels and non-audio features such as HTML pages displayed in synchronization with the audio. RealAudio players are available for Microsoft Windows, the Macintosh, and several UNIX platforms.
- **Winplay** - Winplay offers a very high quality audio using MPEG Level 3 compression. To the best of my knowledge, this feature is not available in any other similar product out there. Unfortunately, it is available for Windows 3.x only. You can download it from URL: <ftp://ftp.uoknor.edu>, or from the Institute for Integrated Circuits home page, in Germany at URL: <http://www.iis.fhg.de/departs/amm/layer3/winplay3>.
- **VocalTec** - This is a well known player, which offers streaming audio technology for the Web, but it is available for Microsoft Windows only. You can check their URL at <http://www.vocaltec.com>

Multicast packets are designated with a special range of IP addresses: 224.0.0.0 to 239.255.255.255. This range, as discussed above, is specifically known as "Class D Internet Addresses". The Internet Address Number Authority (IANA) has given the MBONE (which is largely used for teleconferencing) the Class D subset of 224.2.\*.\*. Hosts choosing to communicate with each other over MBONE set up a session using one IP address from this range. Thus, multicast IP addresses are used to designate a group of hosts attached by a communication link rather than a group connected by a physical LAN. Also, each host temporarily adopts the same IP address. After the session is terminated, the IP address is restored to the "pool" for re-use by other sessions involving different hosts.

There are still some problems to be resolved before MBONE can be fully implemented on the Internet. Since multicasts between multiple hosts on different subnets must be physically transmitted over the Internet and not all routers are capable of multicasting, the multicast IP packets must be tunneled (which makes MBONE a virtual network) to look like unicast packets to ordinary routers. Thus, these multicast IP datagrams must be first encapsulated by the sources-end mrouter in a unicast IP header that has the destination and source address fields set to the IP addresses of tunnel-end-point mrouter respectively and the protocol field set to "IP" which indicates that the next protocol in the packet is also IP. The destination mrouter then strips of this header and reads the "inner" multicast session IP address and forwards the packet to its own network hosts or re-encapsulates the datagram and forwards it to other mrouter that serve or can forward to session group members.

### Note:

For more information about MBONE, check Vinay Kumar book "MBONE: Interactive Multimedia on the Internet," by New Riders, 1996.



# Internet Control Message Protocol (ICMP)

The Internet Control Message Protocol, as defined on RFC 792, is a part of IP that handles error and system level messages and sends them to the offending gateway or host. It uses the basic support of IP as if it were a higher level protocol, however, ICMP is actually an integral part of IP, and must be implemented by every IP module.

Messages are sent in several situations. It could be sent when a datagram does not reach its destination or when a gateway fails to forward a datagram (usually due to not enough buffering capacity), for example.

# Internet Group Management Protocol (IGMP)

Internet Group Management Protocol (IGMP), as defined in RFC 1112, was developed for hosts on multi-access networks to instruct local routers of their group membership information, which is performed by hosts multicasting IGMP Host Membership Reports. These multicast routers listen for these messages and then can exchange group membership information with other multicast routers, which allows distribution trees to be formed to deliver multicast datagrams.

There have been few extensions, known as IGMP version 2, that were developed and released in later releases of the IP Multicast distribution to include explicit leave messages for faster pruning and multicast traceroute messages. Figure 1.17 shows the header information of an IGMP message.

A typical IGMP statement looks like this,

```
igmp  yes |  no |  on |  off [  {

    queryinterval  sec  ;

    timeoutinterval  sec  ;

    interface  interface_list  enable |  disable;

    traceoptions  trace_options  ;

} ]  ;
```

The `igmp` statement on the first line enables or disables the IGMP protocol. If the `igmp` statement is not

specified the default is igmp off; If enabled, IGMP will default to enabling all interfaces that are both broadcast and multicast capable. These interfaces are identified by the IFF\_BROADCAST and IFF\_MULTICAST interface flags. IGMP must be enabled before one of the IP Multicast routing protocols are enabled.

**Note:**

For complete information about IGMP functionality and options, please check RFC 1112 or InterGate's URL at <http://intergate.ipinc.com/support/gated/new/node29.html>

## Open Shortest-Path First (OSPF)

Open Shortest-Path First (OSPF) is a second-generation standards-based IGP (Interior Gateway Protocol) that enables routers in an autonomous system to exchange routing information. By autonomous system I mean those systems that consists of a group of routers under the administrative control of one authority. OSPF minimizes network convergence times across large IP internetworks.

OSPF should not be confuse with RIP as it is not a distance vector routing protocol. Rather, OSPF is a link state routing protocol, permitting routers to exchange information with one another about the reachability of other networks and the cost or metric to reach the other networks. OSPF is defined as one of the IGP standard defined in RFC 1247.

**Tip:**

What is IGP anyway?

Interior Gateway Protocol (IGP) is an Internet protocol designed to distribute routing information to the routers within an autonomous system. To better understand the nature of this IP protocol just substitute the term "gateway" in the name, which is more of a historical definition, with the term "router," which is much more accurate and preferred term.

All routers supporting OSPF exchange routing information within an autonomous system using a link-state algorithm by issuing routing update messages only when a change in topology occurs. In this case, the affected router immediately notifies its neighboring router about the topology change only, instead of the entire routing table. By the same token, the neighbor router pass the updated information to their neighboring routers, and so on, reducing the amount of traffic on the internetwork. The major advantage of this is that since topology change information is propagated immediately, all network convergence is achieved more quickly than if relying on the timer-based mechanism used with RIP.

Hence, OSPF is increasingly being adopted within existing autonomous systems that previously relied on RIP's routing services, especially because OSPF routers simultaneously support RIP for router-to-endstation communications, and OSPF for router-to-router communications. This is great because it ensures communications within an internetwork and provides a smooth migration path for

introducing OSPF into existing networks.

## Border Gateway Protocol Version 4 (BGP-4)

Border Gateway Protocol Version 4 (BGP-4) is an exterior gateway protocol that enables routers in different autonomous systems to exchange routing information. BGP-4 also provides a set of mechanisms for facilitating CIDR by providing the capability of advertising an arbitrary length IP prefix and thus eliminating the concept of network "class" within BGP.

BGP uses TCP to ensure delivery of interautonomous system information. Update messages are generated only if a topology change occurs and contain information only about the change. This reduces network traffic and bandwidth consumption used in maintaining consistent routing tables between routers.

## Address Resolution Protocol

Address Resolution Protocol (ARP) is a method for finding a host's Ethernet address from its Internet address. The sender broadcasts an ARP packet containing the Internet address of another host and waits for it to send back its Ethernet address. Each host maintains a cache of address translations to reduce delay and loading. ARP allows the Internet address to be independent of the Ethernet address but it only works if all hosts support it.

As it is defined on RFC 826, a router and host must be attached to the same network segment to accomplish ARP, and the broadcasts cannot be forwarded by another router to a different network segment.

## *Reverse Address Resolution Protocol (RARP)*

Reverse Address Resolution Protocol (RARP), as defined on RFC 903, provides the reverse function of ARP discussed above. RARP maps a hardware address, also called MAC address, to an IP address. RARP is primarily used by diskless nodes, when they first initialize, to find their Internet address. Its function is very similar to BOOTP.

## *Security Risks of Passing IP Datagram Through Routers*

Routers are often overlooked when dealing with network security. They are the lifeblood of an Internet connection. They provide all the data on a network a path to the outside world. This also makes them a wonderful target for attacks. Since most sites have one router to connect to the outside world, all it takes is one attack to cripple that connection.

Always keep up with the latest version of the router's software. The newer releases can fix a great deal of recently emerged denial-of-service attacks. These attacks are often trivial to execute and require only a

few packets across the connection to trigger. A router upgrade will sometimes mean further expense in memory or firmware upgrades, but as a critical piece of equipment, it should not be neglected.

Other than updating the software, disabling remote management is often key to preventing both denial-of-service attacks and remote attacks to try to gain control of the router. With a remote management port open, attackers have a way into the router. Some routers fall victim to brute-force attempts against their administrative passwords. Quick scripts can be written to try all possible password combinations, accessing the router only once per try to avoid being detected. If there are so many routers that manual administration is a problem, then perhaps investigating network switch technology would be wise. Today's switches are replacing yesterday's routers in network backbones to help simplify such things.

## Simple Network Management Protocol (SNMP)

Simple Network Management Protocol (SNMP), as defined in STD 15, RFC 1157, was developed to manage nodes on an IP network.

One element of IP security that has been somewhat neglected is protection of the network devices themselves. With the Simple Network Management Protocol version 2 (SNMPv2) the authentication measures for management of network devices were strengthened. But based on few controversies, there is an indication that successful incorporation of strong security features on SNMP will take some time.

### **Note:**

Many of the original proposed security aspects of SNMPv2 were made optional or removed from the Internet Standards track SNMPv2 specification in March 1996. There is now a new experimental security protocol for SNMPv2 that has been proposed.

Nevertheless, SNMP is the standard protocol used to monitor and control IP routers and attached networks. This transaction-oriented protocol specifies the transfer of structured management information between SNMP managers and agents. An SNMP manager, residing on a workstation, issues queries to gather information about the status, configuration, and performance of the router.

## Watch Your ISP Connection.

When shopping for an Internet Service Provider, most people glaze over the security measures that are offered to people that subscribe to their service. Their level of security can quickly decide a customer's level of security. If the upstream feed is compromised, then all of the data bound for the Internet can be sniffed by the attacker. It is actually very surprising to see what information is sent back and forth from a customer. Private e-mail can be read. Web form submissions can be read. Downloaded files can be intercepted. Anything that heads for the Internet can be stolen.

There has even been a nasty trend of not just stealing information, but of hijacking connections. A user logs into their remote account and suddenly their files start changing. Hijacking has become quite

advanced. A session can be transparently hijacked and the user will simply think that the network is lagging. Such hijacking does, however, require that the attacker be in the stream somewhere and an ISP is a wonderful place to perch.

## The Internet Protocol Next Generation or IPv6

Since the introduction of TCP/IP to the ARPANET in 1973, which, at that time connected about 250 sites and 750 computers, the Internet has grown tremendously, connecting today more than 60 million users worldwide. Current estimates project the Internet as connecting hundreds of thousands of sites and tens of millions of computers. This phenomenal growth is placing an ever-growing strain on the Internet's infrastructure and underlying technology.

Due to this exponential growth of the Internet, underlying inadequacies in the network's current technology has become more and more evident. The current Internet Protocol version 4 (IPv4) was last revised in 1981 (RFC791), and since then the Engineering Task Force (IETF) has been developing solutions for inadequacies that emerged as the protocol grows old. These sets of solutions, which have been given the name IPv6, will become the backbone for the next generation of communication applications.

It is anticipated that in the early XXI century, just around the corner, the Internet will be routinely used in ways just as unfathomable to us, today. Its usage is expected to extend to multimedia notebook computers, cellular modems, and even appliances at home, such as your TV, your toaster and coffee maker (remember that IBM's latest desktop PC model already comes with some of these remote functionality to control your appliances at home!).

Virtually all the devices with which we interact, at home, at work, and at play, will be connected to the Internet – the possibilities are endless, and the implications staggering, especially as far as security and privacy goes.

To function within this new paradigm TCP/IP must evolve and expand its capabilities, and the first significant step in that evolution is the development of the next generation of the "Internet Protocol," Internet Protocol version 6, or IPv6.

The advent of the Ipv6 initiative doesn't mean that the technologies will exhaust the capabilities of Ipv4, our Internet technology. However, as you might expect, there are still compelling reasons to begin adopting IPv6 as soon as possible. However, this process has its challenges, and as essential to any evolution of Internet technology, there are requirements for seamless compatibility with IPv4, especially with regards to a manageable migration, which would allow us to take advantage of the power of IPv6, without forcing the entire Internet to upgrade simultaneously.

## *Address Expansion*

One of the main needs for IPv6 is the rapid exhaustion of the available IPv4 network addresses. To assign a network address to every car, machine tool, furnace, television, traffic light, EKG monitor, and telephone, we will need hundreds of millions of new network addresses. IPv6 is designed to address this problem globally, providing for billions of billions of addresses with its 128bit architecture.

# ***Automatic Configuration of Network Devices***

It is not an easy task to manually configure and manage the huge number of hosts connected to many networks, public or private. Managers of major corporate networks, as well as Internet Providers are going crazy with it. IPv6's Auto-configuration capability will dramatically reduce this burden by recognizing when a new device has been connected to the network and automatically configuring it to communicate. For mobile and wireless computer users the power of IPv6 will mean much smoother operation and enhanced capabilities.

## ***Security***

At this point on the book, needless to say there is a major security concern shared by senior IT professionals and CEOs when connecting their organization with Intranets and to the Internet. Nevertheless, for everyone connected to the Internet, invasion of privacy is also a concern as IP connections are beginning to invade even coffee makers. Fortunately, IPv6 will have a whole host of new security features built in, including system to system authentication and encryption based data privacy. These capabilities will be critical to the use of the Internet for secure computing.

## ***Real-Time Performance***

One barrier to adoption of TCP/IP for real-time, and near real-time, applications has been the problem of response time and Quality of Service. By taking advantage of IPv6's packet prioritization feature TCP/IP now becomes the protocol of choice for these applications.

## ***Multicasting***

The designs of current network technologies were based on the premise of one-to-one or one-to-all communications. This means that applications that are distributing information to a large number of users must build a separate network connection from the server to each client. IPv6 provides the opportunity to build applications that make much better use of server and network resources through its "multicasting" option. This allows an application to "broadcast" data over the network, where it is received only by those clients that were properly authorized to do so. Multicast technology opens up a whole new range of potential applications, from efficient news and financial data distribution, to video and audio distribution, etc.

There are many features and implementations to be discussed about IPv6, but for our purpose here, let's concentrate on the IPv6's promises, specifically with regards to security.

## ***IPv6 Security***

Users want to know that their transactions and access to their own sites are secure. Users also want to increase security across protocol layers. Up until IPv6, as discussed throughout this whole book, security

has been available only by added applications or services.

IPv6 provides security measures in two functional areas:

- **Authentication** – It requires that a sender log into the receiver. If the sender is not recognized, then access is not allowed. If access is allowed, this ensures that the packets were actually sent by the approved sender and that the content was not changed in transit.
- **Privacy** - Privacy takes the form of encryption and protects data from unintended users. Packets that leave a site can be encrypted and packets that enter a site can be authenticated.

Both privacy and authentication can be applied in a "security association." For a one-way exchange between a sender and a receiver, one association is needed; for a two-way exchange, two associations are needed. When combining authentication and privacy, either can be applied first. If encryption is applied first, the entire packet is authenticated, including encrypted and unencrypted parts. If authentication is applied first, authentication applies to the entire packet.

IPv6 is being tested over and over by IETF and its participating partners. With its core specifications finalized, IPv6 implementations should occur within a year and Internet Service Providers should begin to offer IPv6 links during the next three to four years.

### Tip:

For more up to date information, check the IPv6 Resource Center of Process Software Corporation, one of the leaders in TCP/IP solutions, at URL **Error! Reference source not found..**

## Network Time Protocol (NTP)

Network Time Protocol (NTP) is a protocol built on top of TCP/IP that assures accurate local timekeeping with reference to radio, atomic or other clocks located on the Internet. This protocol is capable of synchronizing distributed clocks within milliseconds over long time periods. It is defined in STD 12, RFC 1119.

## Dynamic Host Configuration Protocol (DHCP)

Dynamic Host Configuration Protocol (DHCP) was actually a protocol introduced by Microsoft on their NT server with version 3.5 in late 1994. This protocol provides a means to dynamically allocate IP addresses to IBM PCs running on a Microsoft Windows local area network.

The system administrator assigns a range of IP addresses to DHCP and each client PC on the LAN has its TCP/IP software configured to request an IP address from the DHCP server. The request and grant process uses a lease concept with a controllable time period. More information can be found in the Microsoft documentation on NT Server.

# Windows Sockets (WINS)

WINS, or Winsock, is a specification for Microsoft Windows network software, describing how applications can access network services, especially TCP/IP. Winsock is intended to provide a single API to which application developers should program and to which multiple network software vendors should conform. For any particular version of Microsoft Windows, it defines a binary interface (ABI) such that an application written to the Windows Sockets API can work with a conformant protocol implementation from any network software vendor.

Windows Sockets is supported by Microsoft Windows, Windows for Workgroups, Win32s, Windows 95 and Windows NT. It also supports protocols other than TCP/IP.

## Domain Name System (DNS)

Domain Name System (DNS), is defined on RFCs 1034 and 1035, is a general-purpose distributed, replicated, data query service chiefly used on Internet for translating hostnames (or site name) such as "process.com" into its IP address such as 192.42.95.1. DNS can be configured to use a sequence of name servers, based on the domains in the name being looked for, until a match is found.

DNS is usually installed as a replacement for the hostname translation offered by Sun Microsystem's Network Information System (NIS). However, while NIS relies on a single server, DNS is a distributed database. It can be queried interactively using the command `nslookup`.

The Domain Name System refers to both the way of naming hosts and the servers and clients that administer that information across the Internet.

## *Limiting DNS Information*

InterNIC holds information about a site's primary and secondary DNS. It is typical to foreign users to refer to InterNIC to learn which system to access to translate addresses into machine names. Be careful which addresses are supplied in the external primary and secondary DNS. Listing vital internal resources in the DNS records, that foreign users can access, can be pointers to determine which systems should be attacked. Externally naming a system "main-server" or "modem-dialout" can be tragic.

Therefore, I suggest you to setup a third DNS server to host internal addresses. Only allow systems from the local site to access this information. This will prevent internal names from being leaked to the Internet. Two different names can be given to hosts that are accessible by the Internet. Internally naming a vital system "main-server" is acceptable if the external name for the system is something less obvious or a limited version of what it hosts, like "ftp" or "www". If there are a lot of machines, it could easily be that only a few systems should be listed externally.



# Firewalls Concepts

By now, only with an overview of internetworking protocols and standards, you should assume that every piece of data sent over the Internet can be stolen or modified. The way the Internet is organized, every site takes responsibility for their own security. If a hacker can take a site that is at a critical point to the communications that are being sent from the user, then all of the data that the user is sending through that site is completely at the whim of the hacker. Hackers can intercept unencrypted credit cards, telnet sessions, ftp sessions, letters to Grandma, and just about anything else that comes across the wire.

Just like not trusting your upstream feed, be careful with the information that is sent to remote sites. Who controls the destination system should always be in question.

Firewalls are designed to keep unwanted and unauthorized traffic from an unprotected network like the Internet out of a private network like your LAN or WAN, yet still allowing you and other users of your local network to access Internet services. Figure 1.18 shows the basic purpose of a firewall.

## ***Figure 1.18***

*Basic function of a firewall*

Most firewalls are merely routers, as showing on figure 1.19, filtering incoming datagrams based upon the datagrams source address, destination address, higher level protocol, or other criteria specified by the private network's security manager, or security policy.

## ***Figure 1.19***

*Packet filtering at a router level*

More sophisticated firewalls employ a proxy server, also called a bastion host, as shown on figure 1.20. The bastion host prevents direct access to Internet services by the your internal users, acting as their proxy, while filtering out unauthorized incoming Internet traffic.

## ***Figure 1.20***

*Proxy server prevent the direct access to and from the Internet.*

The purpose of a firewall, as a security gate, is to provide security to those components inside the gate, as well as control of who (or what) is allowed to get into this protected environment, as well as those allowed to go out. It works like a security guard at a front door, controlling and authenticating who can or cannot have access to the site.

It is setup to provide controllable filtering of network traffic, allowing restricted access to certain Internet port numbers and blocked access to almost everything else. In order to do that, they must function as a single point of entry. That is why many times you will find firewalls integrated with routers.

Therefore, you should choose your firewall system based on the hardware you already have installed at your site, the expertise you have available in your department and the vendors you can trust.

**Note:**

Such is the need for firewalls that according to the journal CommunicationsWeek (April 8, 1996), the Computer Security Institute, from San Francisco-CA, did a survey last year, and found that almost half of the organizations surveyed already deploy firewall, and of those that did not, 70 percent were planning on installing them.

Usually, firewalls are configured to protect against unauthenticated interactive login from the "outside" world. Protecting your site with firewalls can be the easiest way to promote a "gate" where security and audit can be imposed.

With firewalls you can protect your site from arbitrary connections and can even set up tracing tools, which can help you with summary logs about origin of connections coming through, the amount of traffic your server is serving and even if there were any attempts to break in to it.

One of the basic purposes of a firewall should be to protect your site against hackers. As discussed earlier, your site is exposed to numerous threats, and a firewalls can help you. However, it cannot protect you against connections by-passing it. Therefore, be careful with backdoors such as modem connections, to your LAN, specially if your Remote Access Server (RAS) is inside the protected LAN, as typically they are.

Nevertheless, a firewall is not infallible, its purpose is to enhance security, not guarantee it! If you have very valuable information in your LAN, your Web server should not be connected to it in the first place. You must be careful with groupware applications that allows you access to your Web server from within the organization or vice versa.

Also, if you have a Web server inside your internal LAN, watch for internal attacks as well as to your corporate servers. There is nothing a firewall can do about threats coming from inside the organization. An upset employee, for example, could pull the plug of your corporate server, shutting it down, and there is nothing a firewall will be able to do!

Packet filtering was always a simple and efficient way of filtering inbound unwanted packets of information by intercepting data packets, reading them, and rejecting those not matching the criteria programmed at the router.

Unfortunately, packet filtering are no longer sufficient to guarantee the security of a site. Many are the threats, and many are the new protocol innovations, with the ability to by-pass those filters with very little efforts.

For instance, packet filtering is not effective with the FTP protocol as FTP allows the external server being contacted to make connections back on port 20 in order to complete a data transfers. Even if a rule were to be added on the router, port 20 on the internal network machines is still available to probes from the outside. Besides, as seen earlier, hackers can easily "spoof" these routers. Firewalls make these strategies a bit harder, if not near to impossible.

When deciding to implement a firewall, however, first you will need to decide on the type of firewall to be used (yes, there are many!) and its design. I'm sure this book will greatly help you in doing so!

You should also know that there are some kind of commercial firewall products, often called OS Shields, that are installed over the operating system. Although they became some what popular, combining packet filtering with proxy applications capable to monitor data and command streams of any protocol to secure the sites, OS shield was not so successful due to specifics of its configurations: not only its configurations were not visible to administrators as they were configured at the kernel level, but also forced administrators to introduce additional products to help the management of the server's security.

The firewall technology has gone a long way. Besides the so called traditional, or static, firewalls, today we have what is called "dynamic firewall technology."

The main difference is that, unlike static firewalls, where the main purpose is to

**"permit any service unless it is expressly denied" or to**

**"deny any service unless it is expressly permitted,"**

a dynamic firewall will

**"permit/deny any service for when and as long as you want."**

This ability to adapt to network traffic and design offers a distinctive advantage over the static packet filtering models.

## ***The Flaws in Firewalls***

As you can tell by the number of pages of this book (and we're still on chapter 1!), there is a lot to be said about firewalls, especially because virtually all of the latest generation of firewalls exhibit the same fundamental problems: they can control which site can talk to which services at a certain time and only if a certain authorization is given, but services that are offered to the Internet as a whole can be shockingly open!

The one things firewalls cannot currently do is understand the data that goes through to a valid service. To the firewall, an e-mail message is an e-mail message. Data filtering is a recent invention in some firewalls, for more information check chapter 10 "Putting it Together: Firewall Design and Implementation," under the section "Types and Models of Firewalls."

To have a firewall filter and remove every message with the word "hacker," for example, is already possible, but not all of them have the ability to filter applets, which is nowadays a major threat to any protected corporate network.

Also, if a hacker connects to a valid service or port on a system inside a firewall, such as the SMTP port, then the hacker can use a valid data attack, or shell commands, to exploit that service.

Take a Web server as an example. One of the most recent attacks against NCSA Web servers is the 'phf' attack. A default utility 'phf' comes with the server and allows an attacker to use the utility to execute commands on the systems. The attack looks like a normal Web query. Today's firewalls will not stop this

attack, unless an administrator mail-filters on 'phf', which places a high demand on the firewall.

The key to dealing with this limitation is in treating a firewall as a way of understanding the configuration of internal services. The firewall will only allow certain services to be accessed by users on the Internet. These known services can then be given special attention to make sure that they are the latest, most secure versions available. In this way, the focus can shift from hardening an entire network, to just hardening a few internal machines and services.

More will be seen about it on chapters 4 "Firewalling Challenges: the Basic Web," chapter 5 "Firewalling Challenges: the Advanced Web," and chapter 8 "How Vulnerable are Internet Services."

## ***Fun With DMZs***

Demilitarized Zones (DMZs) are used in situations where few machines service the Intranet and the rest of the machines are isolated behind some device, usually a firewall. These machines either sit out in the open or have another firewall to protect the DMZ. This can be a very nice arrangement, from a security perspective, as the only machines that accept inbound connections are "sacrificial lambs."

If the machines can be spared for the effort, organizations that are high risks targets can benefit from this design. It has proven to be extremely effective in keeping internal resources secure. One suggestion is to vary the types of machines and publishers of the security software that guards the outside and the inside of the DMZ. For example, if two of the same firewalls are used, then they can both be breached by one exploit. In a homogenous-leaning community, this is one case that being heterogeneous can help.

The only drawback to setting up a DMZ is in the maintenance of the machines. Most administrators enjoy local access to a file-system for easy Web server and FTP server updates. Adding a firewall between the two makes it slightly harder to accomplish this, especially if more than one person is maintaining the servers. All in all, external information stays somewhat stable and the administrative annoyance can be really infrequent.

## ***Authentication Issues***

Firewalls and filtering routers tend to behave rather binary. Either a connection is or is not allowed into a system. Authentication allows service connections to be based on the authentication of the user, rather than their source or destination address. With some software, a user's authentication can allow certain services and machines to be reached while others can only access rudimentary systems. Firewalls often play a large role in user-based service authentication, but some servers can be configured to understand this information as well. Current Web servers can be configured to understand which users are allowed to access which sub-trees and restrict users to their proper security level.

Authentication comes in many varieties and it can be in the form of cryptographic tokens, one-time passwords, and the most commonly used and least secure simple text password. It is up to the administrators of a site to determine which form of authentication for which users, but it is commonly admitted that it should be used. Proper authentication can allow administrator from foreign sites to come into a network and correct problems. This sort of connection would be a prime candidate for a strong method of authentication like a cryptographic token.

# ***Trust at the Perimeter***

Today's corporate security focus is on the perimeter. It is a very common approach to see a hard coated outside and a soft middle. The hard outside is accomplished with firewalls, authentication devices, strong dial-up banks, virtual private tunnels, virtual networks, and a slew of other ways to isolate a network. The inside, however, is left up for grabs. Internal security is not properly managed and a common looming fear exists that if someone gets past the borders, then the castle will fall. It is often a problem that everyone knows about and is eternally scheduled to be fixed tomorrow.

There really is not a lot to be said for a solution to this problem. The internal politics of security is usually a quagmire of sensitive issues and reluctance to properly fund a solution. The only way that this issue can get solved is through good old fashioned soap-boxing and a fervent interest to help the effort along. Political issues are infrequently solved quickly or permanently, but the truth in trusting a perimeter is one of eventual disappointment.

The issue of breaching firewalls has already been discussed and authentication methods are far from idiot-proof. Trusting the physical security of a site can be just as disastrous. The level of identification required from outsiders is usually horribly inadequate. How often is the telephone repair person checked up on? Would the repair person be given access to the most sensitive parts of an organization? The bottom line is that the perimeter is not the only place for security.

## ***Intranets***

Resources provided by Intranets are rapidly becoming a staple good within information systems groups. They promise to provide a single resource that everyone can access and enrich their lives. Switching to a paperless information distribution system is not always as grand as it looks. Placing all of an organizations internal documentation into one place is akin to waving a giant red flag and expecting people not to notice.

Perhaps I'm creating a new word, but Intra-Intranets are often a wise solution to this issue. Keeping critical data within Workgroup and non-critical data in a separate Intranet is a viable alternative. Use different systems to store subgroups on and one main system for the whole organization. Policies should be developed for what is allowable on the main system to help keep proprietary material away from public or near-public access.

## **From Here...**

This chapter provided a comprehensive overview on many of the most used internetworking protocols and standards, some of the security concerns associated with it and the basic whole of firewalls in enhancing the security of the connections you make across the Internet and receive within your protected network.

The issue of basic connectivity becomes then very important for many organizations. There are indeed many ways to get connected on the Internet, some more effective then others due to their ability to interact with a variety of environments and computers.

Chapter 2, "Basic Connectivity," discuss about the characteristics a basic connectivity can assume on the Internet through UUCP, SLIP, PPP, Rlogin and TELNET.

Orders

Orders

Back

Forward

Comments

Comments

---

[COMPUTING MCG](#)

|        |        |
|--------|--------|
| Orders | Orders |
|--------|--------|

|      |         |
|------|---------|
| Back | Forward |
|------|---------|

|          |          |
|----------|----------|
| Comments | Comments |
|----------|----------|

© 1997 The McGraw-Hill Companies, Inc. All rights reserved.  
Any use of this Beta Book is subject to the rules stated in the [Terms of Use](#).

---

# Chapter 2

## *Basic Connectivity*

As you saw on chapter 1, "Internetworking Protocols and Standards: An Overview," the popularity of TCP/IP and all the standards and protocols derived from it makes the issue of basic connectivity critical for many organizations. As we realize, there are indeed many ways to get connected on the Internet, some more effective than others due to their ability to interact with a variety of environments and computers.

Regardless if you look at the Internet as a verb, as internetworking couple LANs or WANs, or you use it as a noun, as comprising two or more different networks, you will have to get down to basics when talking about connectivity and how you will connect clients, servers and networks (LAN and WAN), and ultimately, how you will protect this connections. We also discussed on chapter 1 about the many protocols used and in use on the Internet, as well as those being developed and proposed (IPv6). But can your organization take advantage of the Internet, or the IP technology for that matter? What kind of topology you have in your company? How is file transfer, electronic mail, host terminal emulation sessions, hardware integration and most of all, security, handled at your company?

These are issues that you need to have clear in your mind so that you can communicate with management and MIS in your company and to focus in the technology that you need to effectively deploy your basic connectivity plan, no matter with you are just starting or have a large and complex network system. When internetworking, you must keep the focus of secure connections and how you intend to deploy it.

TCP/IP technology will provide the basic connectivity that is needed within any organization as it collects, analyses, and distributes information. Advanced knowledge of rapidly evolving storage technologies will always be essential to accommodate voice, video, and other broad bandwidth sources of information. But you must be ready to chose and deploy the right protocol, the right technology, for the kind of connectivity you need securely, after all, the Internet, as it is discussed later in more details, is a wild place!

Being the Internet basically a virtual network that allows users to communicate with all connected servers and hosts, as if they were part of a local network, then there is a need for all details of this network to be hidden from all users. This is were the basic connectivity requirements starts, and the basis on which this virtual network exists is actually provided by the TCP/IP suite. The many protocols, as

seen on Chapter 1, establishes the format and the rules that must be followed for information to be exchanged between systems. But how are services to network users provided and what are the security issues surrounding it?

TCP/IP defines a wide range of application layer protocols that provide services to network users, including remote login, file copying, file sharing, electronic mail, directory services, and network management facilities. Some application protocols are widely used, others are employed only for specialized purposes. Although throughout this chapter we will only concentrate on some of these protocols and their security weaknesses, the following are the most commonly used TCP/IP application layer protocols:

- **PING** - According to the Computer Dictionary (<http://nightflight.com/foldoc/>), PING was probably originally contrived to match submariners' term for the sound of a returned sonar pulse! But actually, this is a program used to test network connectivity by sending them one, or repeated, ICMP echo requests and waiting for replies. Since ping works at the IP level its server-side is often implemented entirely within the operating system kernel and is thus pretty much the lowest level test of whether a remote host is alive. Ping will often respond even when higher level, TCP-based services cannot.
- **TELNET** - This is the Internet standard protocol for remote login. It runs on top of TCP/IP.
- **Rlogin** - Similar to TELNET, Rlogin is the 4.2BSD UNIX utility to allow a user to log in on another host via a network. Rlogin communicates with a daemon on the remote host.
- **Rsh** - The acronym stands for "Remote shell." This is a Berkeley UNIX networking command to execute a given command on a remote host, passing it input and receiving its output. Rsh communicates with a daemon on the remote host.
- **FTP** - Acronym for File Transfer Protocol, this is a client-server protocol that enables the file transferring between two computers over a TCP/IP network.
- **TFTP** - Acronym for Trivial File Transfer Protocol, this is a very similar to FTP, this is a simple file transfer protocol usually used for down-loading boot code to diskless workstations.
- **SMTP** - Acronym for Simple Mail Transfer Protocol, this protocol is used to transfer electronic mail between computers.
- **Kerberos** - This is an authentication system developed at MIT, based on symmetric key cryptography.
- **X Windows** - A specification for device-independent windowing operations on bitmap display devices.
- **DNS Name** - A general-purpose distributed, replicated, data query service chiefly used on Internet for translating hostnames into Internet addresses.
- **NFS** - Acronym for Network File System, a protocol that allows a computer to access files over a network as if they were on its local disks.
- **SNMP** - Acronym for Simple network Management Protocol, which is the Internet standard protocol to manage nodes on an IP network.



# ***What Happened to TTY***

TTY is actually a TeleTYpe terminal, characterized by a noisy mechanical printer, a very limited character set, and poor print quality.

But especially in the UNIX world, TTY is characterized as any terminal at all. The term can be used to refer to the particular terminal controlling a given job, besides also being the name of a UNIX command which outputs the name of the current controlling terminal! Also, the term can refer to any serial port, whether or not the device connected to it is a terminal. I think this happens because under UNIX such devices have names of the form `tty*`. Yes the term has some ambiguity, so does the way it is still being used nowadays

For the purpose of focusing on basic connectivity its main usability nowadays, lets stick to TTY as being a device that allows text messages to be converted into speech and vice-versa. Also called a Telecommunications Device for the Deaf (TDD), this is a terminal device used widely by deaf people for text communication over telephone lines.

Usually, a relay operator is necessary to interact between a sender or receiver without a TTY modem. If someone needs to communicate with a person who has a TTY, the Relay operator will type in what the user is saying on his/her TTY. The information is then "relayed" to the caller without a TTY and vice versa.

The major difference between a TTY/TDD device and a regular modem is that a TTY uses the BAUDOT coding to communicate and a typical modem will use ASCII. BAUDOT coding is not new and also supports a very limited character number of character set. That is one reason modem manufacturers migrated to ASCII. Also, these device communicate at a very low speed, about 300 or less baud. Most of TTYs in US communicate at 45.45 bits per second.

Although standard modems are no compatible to TTY, there are some of TTY modems that support ASCII communications. But again, you may have problems suing them as most of the TTY modems will communicate at a maximum speed of 300 baud, some being as low as 110.

By the same token, there are modems that will do the ASCII to BAUDOT conversion allowing you a decent level of conversion.

## ***What is the Baudot Code?***

Extensively used in telegraph systems, the Baudot code was invented by the Emile Baudot in 1870. This asynchronous code uses only five bits, allowing up to 32 different characters. To accommodate all the letters of the alphabet and numerals, two of the 32 combinations were used to select alternate character sets. Table 2.1 shows a list of all the possible characters available in Baudot.

**Note:**

There is a software, as show on the screenshot of figure 2.1, that allows you a modem to talk with a TTY modem that has the ASCII mode option turned on. You can find more information about it at the URL <http://tap.gallaudet.edu/asciitdd.htm>.

**Table 2.1 - All Possible Characters in Baudot**

| Binary | Hex | LTRS | FIGS |
|--------|-----|------|------|
| 00011  | 03  | A    | -    |
| 11001  | 19  | B    | ?    |
| 01110  | 0E  | C    | :    |
| 01001  | 09  | D    | \$   |
| 00001  | 01  | E    | 3    |
| 01101  | 0D  | F    | !    |
| 11010  | 1A  | G    | &    |
| 10100  | 14  | H    | #    |
| 00110  | 06  | I    | 8    |
| 01011  | 0B  | J    | BELL |
| 01111  | 0F  | K    | (    |
| 10010  | 12  | L    | )    |
| 11100  | 1C  | M    | .    |
| 01100  | 0C  | N    | ,    |
| 11000  | 18  | O    | 9    |
| 10110  | 16  | P    | 0    |
| 10111  | 17  | Q    | 1    |

|       |    |        |        |
|-------|----|--------|--------|
| 01010 | 0A | R      | 4      |
| 00101 | 05 | S      | ‘      |
| 10000 | 10 | T      | 5      |
| 00111 | 07 | U      | 7      |
| 11110 | 1E | V      | ;      |
| 10011 | 13 | W      | 2      |
| 11101 | 1D | X      | /      |
| 10101 | 15 | Y      | 6      |
| 10001 | 11 | Z      | "      |
| 01000 | 08 | CR     | CR     |
| 00010 | 02 | LF     | LF     |
| 00100 | 04 | SP     | SP     |
| 11111 | 1F | LTRS   | LTRS   |
| 11011 | 1B | FIGS   | FIGS   |
| 00000 | 00 | Unused | Unused |

Where ‘CR’ is carriage return, ‘LF’ is linefeed, ‘BELL’ is the bell, ‘SP’ is space, and ‘STOP’ is the stop character.

## ***UNIX to UNIX CoPy (UUCP)***

The UNIX to UNIX CoPy (UUCP) is the built-in networking system that comes with every UNIX system, which is basically used to provide access to the Internet off-line. UUCP features are limited in many ways, only allowing the exchange of messages and not providing a remote login facility as TCP/IP does. However, it is still very common among the BBS’s (Bulletin Board System) to allow users to have access to electronic mail, although this feature is very slow and awkward if compared to TCP/IP-based systems.

Now, within UUCP there is actually very simple program called "uucp." The basically function of uucp

is to copy files from one host to another, but it also allows certain actions to be performed on the remote host. Just make sure not to confuse UUCP with uucp, as the first was named after the later, but are not the same thing.

**Tip:**

These are the key UUCP programs:

**uucp** - allows the request of file transferring between remote machines

**UUX** - request command execution on remote machines, mail transfers

**UUXQT** - process remote requests locally, both uucp and UUX, running on background.

**UUCICO** - calls and transfers files and requests by UUCP and UUX. Master/Slave configuration

Another property of UUCP is that it allows to forward jobs and files through several hosts, through a chain, provided they cooperate. The most important services provided by UUCP networks these days are electronic mail and news.

Finally, UUCP is also the medium of choice for many dial-up archive sites which offer public access. You can usually access them by dialing them up with UUCP, logging in as a guest user, and download files from a publicly accessible archive area. These guest accounts often have a login name and password of uucp/nuucp or something similar.

## ***SLIP and PPP***

SLIP or Serial Line Internet Protocol is a communications protocol that supports an Internet connection (i.e., using TCP/IP) over a dial-up line using an RS-232 serial port connected to a modem.

SLIP modifies a standard Internet datagram By appending a special SLIP END character an Internet datagram, SLIP modifies the datagram, which allows it to be distinguished as separate. SLIP requires a port configuration of 8 data bits, no parity, and hardware flow control. However, SLIP does not provide error detection, and relies on other high-layer protocols for this. Over a particularly error-prone dial-up link therefore, SLIP on its own would not be satisfactory.

In order to work properly, a SLIP connection must have its IP address configuration set each time before it is established.

As for PPP or Point-to-Point Protocol, is a newer protocol that does essentially the same thing SLIP does. However it's better designed and more acceptable due to its advantages over SLIP. PPP has a number of advantages over SLIP through its design to operate both over asynchronous connections and bit-oriented synchronous systems, and the ability to configure connections to a remote network dynamically, as well as test a link is for connectivity.

**NOTE:**

PPP can be configured to encapsulate different network layer protocols (such as IP, IPX, or AppleTalk) by using the appropriate Network Control Protocol (NCP). For more information, check the URL <http://www.virtualschool.edu/mon/DialupIP/slip-ppp.html>

## ***Rlogin***

Just like TELNET, rlogin connects your terminal on the current local host system lhost to the remote host system rhost.

Cygnus Solutions (<http://www.cygnus.com>) has a product called KerbNet, as shown on the screenshot of their site on figure 2.2, that enables a very secure connection using rlogin.

The version built to use Kerberos authentication is very similar to the standard Berkeley rlogin, except that instead of the `rhosts' mechanism, it uses Kerberos authentication to determine whether a user is authorized to use the remote account.

Each user may have a private authorization list in a file `.klogin' in his login directory. This file functions much like the `rhosts' file, by allowing non-local users to access the Kerberos service on the machine where the `.klogin' file exists. For example, user `joe@EAT.COM' would normally not be permitted to log in to machines in the `MUSSELS.COM' realm. However, Joe's friend `bertha@MUSSELS.COM' can create a `.klogin' file in her home directory, that contains the line `joe@EAT.COM'. This allows Joe to log in as Bertha to Bertha's machine, even though does not have a ticket identifying him as Bertha.

Each line in this file should contain a Kerberos principal name of the form `principal.instance@realm'. The following are all valid Kerberos principal names.

If the originating user is authenticated to one of the principals named in `.klogin', access is granted to the account. The principal `accountname@localrealm' is granted access if there is no `.klogin' file.

Otherwise, a login and password is prompted for on the remote machine as in login. To avoid security problems, the `.klogin' file must be owned by the remote user.

If there is some problem in gathering the Kerberos authentication information, an error message is printed and the standard UCB rlogin is executed in place of the Kerberos rlogin. This permits the use of the same rlogin command to connect to hosts that do not use CNS, as well as to host which do.

## ***Virtual Terminal Protocol (TELNET)***

The TELNET protocol is probably on of the most used application protocol to log onto other hosts to obtain or exchange information. Both computers involved in the connection must use/support the TELNET protocol in order for TELNET to work. The computer that you are connecting to via TELNET will usually prompt you for a username and password; if you are not connecting to a public or general account, you will need to have your own account set up prior to your login.

There are few connection-oriented security requirements that you should be aware of when TELNETing:

- Confidentiality
- Integrity
- Peer-entity authentication
- Identity-based access control

All these requirements implicitly assume that there is a basic security implemented at a connection level, stream-oriented, using point-to-point application protocol. But you can not assume that the connection is secure, as not always you will find security mechanisms implemented within the application protocols. If necessary, you must try to implement security mechanisms at lower layers, such as the transport or the network layers.

The Transport Layer Security Protocol (TLSP), which became an Internet Standard in July of 1992, is a possible solution for the lack of security of TELNET connections. TLSP will run under the transport layer and provide security services to TELNET connections on a per-connection basis by providing end-to-end cryptographic encoding directly above the network layer.

One of the main advantages of relaying on this lower layer security mechanisms is that it can avoid the duplication of security efforts. But again, I'm not sure how many developers or implementation professionals would be willing to introduce new software into operating system kernels. Therefore, you would be better off providing security for TELNET connections at the Application layer than at the Network or Transport layer.

## ***Columbia University's KERMIT: a Secure and Reliable TELNET Server***

Information Systems and Technology has come a long way, but many of the main operating systems (OS) do not provide TELNET features that would make its use and security implementations more reliable or at least available. Windows NT 4.0 does have a TELNET interface, as show on figure 2.3, which does a great job, but ever since Windows 95 came out, the comp.os.ms-windows.win95.\* newsgroups have been flooded with requests for a "TELNET server" or "TELNET daemon" for Windows 95.

Why? There is a great document at Columbia University's Web site, at URL <http://www.columbia.edu/kermit/k95host.html> that discuss this issue and introduce a great product, KERMIT, that does a great job fulfilling the Windows 95/NT user community.

As the article indicates, people who own Windows 95 systems want to be able to grant access to their friends, relatives, co-workers, customers, or clients – and to themselves -- at other locations even when those coming into the Windows 95 system do not have Windows 95, or any other version of Windows for that matter (!), or even a PC. In situations like this, "remote access" solutions like PcAnywhere can not be used.

Meanwhile, others want their friends or customers to be able to dial in (not TELNET) to their Windows 95 PCs, because one party or both are not on the Internet. People also have a need for TELNET server

for the same security reasons outlined above. They want to be able to TELNET to a host, a log in. That is, they need a mechanism to provide some form of authentication and access control -- not just a wide-open DOS prompt.

Columbia University's Kermit 95 has lots of features to aid a TELNET connection, as well as making it more secure and easy to use. Figure 2.4 gives you a good overview of Kermit 95 (K-95). You can see the K-95 Dialer interface in the background, and the Connection one in front of it with the entry settings highlighted, open to its first page, and finally the session itself, a dialup connection to a BBS.

Also, figure 2.4 shows in the background, a second session, this one via Internet to a UNIX server, where a piece of a "man page" is showing, to illustrate how the K-95 Dialer can manage multiple sessions. Usually, all you would have to do to open a session would be to double-click on the desired entry.

Figure 2.5 shows the terminal settings page of the entry notebook. Kermit provides one of these notebooks for each connection, so each one can have a different emulation, character size, character set, screen size, colors, and so on. All these settings apply equally well to dialup connections and to TELNET or RLOGIN sessions, and they are all applied automatically as part of the connection process. These notebooks give you fully customized one-button access to every dialup and Internet service or computer that you use.

Figure 2.6 shows how you can give K-95 the information it needs to place your calls correctly, no matter where you are. You don't have to use any of these features if you always make your calls from the same place, but if you travel around with a laptop, you'll be amazed at the convenience. Just tell Kermit 95 (or Windows 95) your new location, and all the numbers in the dialing directory will "just work".

Another great feature of K95 is that, unlike many computers or TELNET services that require different codes for backspacing (many times you have to assign the appropriate code to your PC's backspace), Kermit 95 allows you to assign for each computer or host in your directory their own key settings, specified on the Keyboard tab of its settings notebook, as shown in figure 2.7.

As also shown on figure 2.7, to solve the Backspace problem, just push the appropriate button. Kermit 95 also allows you to load in an entire custom key map for your whole keyboard if you need to (figure 2.7 shows the Key map for host-based WordPerfect 5.1, which is distributed with Kermit 95).

Figure 2.8 illustrates some of K-95's features, such as,

- **Tall screens** - Did you know that in your TELNET sessions without Kermit 95, the Lynx main page is in a 43-line screen?
- **Multi-sized screens** - Which are based on the size of the fonts,
- **Ability to display Latin-1 8-bit characters** - Figure 2.7 blue screen shows a sample of German font.
- **File transferring** - K-95 can actually achieve great transfer rates using long packets and sliding windows, even when, as showing on figure 2.8, the PC is fairly heavily loaded up with other processes.
- **Simultaneous multi-sessions** - K-95, as you can see on the same figure, can handle various other sessions simultaneously, such as ANSI terminal emulation on a dialup session to a BBS, plus the ability to customize your screen colors for each session.

Figure 2.9 shows various context-sensitive pop-up help windows in the Terminal screen - "Important Keys", mouse buttons, Compose-key functions.

## ***TELNET Services Security Considerations***

Despite products such as Kermit or other security mechanisms you've implemented, there are potential security measures you should be aware of:

- **Time out strategies:**
- **Length of TELNET sessions** - You can setup the duration of your users's TELNET session. The length of time could be based on the type of user or the individual user. For example, guest account using TELNET at your company could have a shorter logon time (5-10 minutes) than technical support, upper management or any other qualified/certified user.
- **Session time-out** - A TELNET session can be setup to time-out if no activity occurs after a specific timeframe.
- **Secure screen savers** - You could use a time out screen saver to go off when no activity occurs in a session after a certain period of time. In this case, unlike session time-out, the TELNET session would remain active on the network, but protected. Users could be warned before time-out occurred.
- **Data protection strategies:**
- **Clearinghouse directories** - You should implement corporate wide temporary directories where unverified data entries are saved. Also you should make sure these data would remain unmodifiable by any unauthorized users after entry verified by electronic signature.
- **Protect sensitive data** - Make sure to protect sensitive data, by only allowing validated users to access it and reminding every user that all data is confidential.

## ***A Systems Manager Approach to Network Security***

When we talk about Internet security, what do mean? Do we mean the processes or we mean the results? Do we mean setting up access control and authorization mechanisms or we mean ensuring that users can only perform tasks they are authorized to do, only obtain information they are authorized to have, and are unable to cause damage to any data, applications and operating areas they have access to? I think the later statement defines more of a security environment then the earlier one.

The issue is that network security calls for protections against malicious attack by hackers and intruders, but security is also associated with controlling and authorization mechanisms and the prevention of the effects of errors and equipment failures.

This section aims to provide you with specific measures you should take to improve the security of your network. Although it won't still guarantee a 100% security of your site, you will need more then that, such as cryptography (see chapter 3 "Cryptography: Is it Enough?") and firewalls, discussed throughout this book, before going into specifics, you must understand who your enemies, your risks and challenges are, as well as what are the basic proactive (not reactive!!) alternatives you can use to prevent a security incident.



## From Who Are You Protecting Your Network?

Yes, of course you're protecting your network from hackers, wackers and crackers! But you should consider who these "bandits" might be and want so you can build your security around it. Thus, you must understand what their motivations are. Also, you must determine what they might want to do and the damage that they could cause to your network.

But keep in mind that even if you were to put all the security measures together, they would never be able to make it impossible for a user to perform unauthorized tasks with a computer system. They can only make it harder. The idea is to make sure the network security controls are beyond either the attacker's ability or motivation.

## Is All the Security Efforts Worth?

Please understand that any security measures you adopt will usually reduce the convenience and accessibility of your services. Security can also delay productivity of your users, as well as systems' processing (and many times dedicated hardware), besides creating an expensive administrative and educational overhead.

Therefore, it's important that when we begin to design our security measures (and we'll do it later on in this book!), you should understand their costs and weigh those costs against the potential benefits. To do that, you must understand the costs of the measures themselves and the costs and likelihood's of security breaches. If you incur security costs out of proportion to the actual dangers, you have done yourself a disservice.

## What does Your Gut Feelings Tell You?

Every security professional, by condition or opinion, has his/her own underlying assumptions. You may feel that you have never been hacked, after all not always you can tell it for sure! Maybe you assume that you know more than any hacker, or that your security is enough. No matter what your assumptions and gut feelings are, be careful! If you leave it alone and do not validate it any assumption can become a potential security hole.

## Watch for Confidentiality

I know you've heard many times already about confidentiality, but if you stop to think about it, most security policies are based on secrets. Your passwords are secret, so are your encryption keys, right?

But how secret is secret? Very often, they aren't! The most important part of keeping secrets is knowing the areas you need to protect and keep secret! Thus, start by asking yourself what knowledge would enable someone to circumvent your system. Once you identify it, be zealous about it! Guard that knowledge and assume that everything else is known to your beloved hackers. But don't go nuts! The more secrets you have, the harder it will be to keep all of them... secret!

Your Internet security policy should be developed so that only a limited number of secrets need to be kept, and a selected number of people would need to know it.

## To Err is Human!

Many security policies fail because they did not consider the human factor. Your users are the ones actually using, enforcing or breaking the security policy, and to them, rules and procedures can be difficult to remember, or they don't feel it makes sense to be generating "nonsense" passwords every six months, and so on.

That's why it is still very common to find passwords written on the undersides of keyboards, modems are connected to a network without any security measures, to avoid onerous dial-in security procedures, and so forth! The bottom line is that, if your security measures interfere with essential use of the system, those measures will be resisted by your users and they WILL circumvent it! To make sure you get the support of your users, must make sure that your security procedures are not getting in the way of their work, that they are still getting their jobs done without stress. You'll need to sell it to them!

Remember that any user can compromise your security policy and statistically speaking, most security break-ins came from inside, which not necessarily mean from your users, but it means that there was a hole in the security from within.

Passwords, for instance, can often be found simply by calling the user on the telephone, claiming to be a system administrator, and asking for them. If your users understand security issues, and if they understand the reasons for your security measures, they are far less likely to make a hacker's life easier. At a minimum, users should be taught never to release passwords or other secrets over unsecured telephone lines (especially cellular telephones) or electronic mail.

## Where is your Achilles Tendon?

Every network and security policy has vulnerabilities. Yours is not an exception. Since weak points of a system are usually the ones exploited, you must be aware of the areas that present a danger to your security and plug the holes right away. Identifying your network's weak points is the first step towards developing a sound and secure network.

## The KISS Principle!

It's all right that you create appropriate barriers around your network to protect it against the wild Internet. But don't forget the KISS (Keep It Simple... Stern!) principle! The security of a system is only as good as the weakest security level of any single host in the system, so understand your environment, how it normally functions, knowing what is expected and what is unexpected, and build your policy around it.

Keep your eye on unusual events. It can help you to catch intruders before they can damage the system. Use auditing tools to help you to detect those unusual events.

## ***TELNET Session Security Checklist***

- Enforce the use of passwords of at least 8 character and force them to be changed every 6 months
- Restrict TELNET sessions and access by password and terminal location

- If TELNET sessions are started from home or any other remote location, by telephone dial-up, you should require a second password or a call-back procedure
- Passwords should be encrypted
- Do not allow the share of passwords!
- Log all access by password and network address and construct reports of usage with user name, network address and date (Access audit trail).
- Develop user profiles and monitor deviations from the profile.
- TELNET users should sign a confidentiality agreement.
- Run security test drills periodically with some available security testing programs, and lastly
- As shown on figure 2.10, implement a firewall!

## Trivial File Transfer Protocol (TFTP)

TFTP provides file transfer capabilities with minimal network overhead. Although TFTP uses UDP to transport files between network devices, it supports time-out and retransmission techniques to ensure data delivery.

One of TFTP main weakness is that it allows unauthorized remote access to system or user files because it provides remote access to files, without asking for a password. That's why TFTP is typically used for the initialization of diskless computers, of X terminals, or of other dedicated hardware. But since the TFTP daemon does not limit access to specific files or hosts, a remote intruder could easily use the service to obtain copies of the password file of other system or user files, or even to remotely overwrite files.

Therefore, my recommendation is that you restrict TFTP access to only limited subdirectory trees in the hard drive of your server. Never allow an user to access the root directory of a server, and restrict TFTP access by using a tcp wrapper.

## *TFTP Security Considerations*

For security's sake, TFTP should not be run, but if you must, then use the secure option/flag to restrict access to a directory that has no valuable information, or run it under the control of a chroot wrapper program.

Another utility you can use is rpcinfo, which can talk to the portmapper and show you if the host is running NIS (and even if it is a NIS server or slave), if a diskless workstation is around, if it is running NFS, and any of the info services (rusersd, rstatd, etc.), as well as any other unusual programs.

Secure RPC can help you a lot in diminish the threat on your remote connections, but it has its own problems as it is difficult to administer and the cryptographic methods available for it are not strong. Yes, I know what you're probably thinking, that NIS+, from Sun, fixes some of these problems. But have you seeing substantial results? Don't forget that NIS+ has been limited to running on Suns only!

The solution? Here we come with packet filtering again, or firewalls if you prefer. If you filter packets

coming on port 111, at the very least, a lot of security incidents can be avoided.

But don't rely only on it. The portmapper only knows about RPC services. Other network services can be located with a brute-force method that connects to all network ports. Many network utilities and windowing systems listen to specific ports, such as port 25 for sendmail, port 23 for TELNET and port 6000 for X windows. SATAN includes a program that scans the ports of a remote hosts and reports on its findings providing outputs like the ones below:

```
hacker % tcpmap poorsite.com
```

```
Mapping 148.158.28.1
```

```
port 21: ftp
```

```
port 23: telnet
```

```
port 25: smtp
```

```
port 37: time
```

```
port 79: finger
```

```
port 512: exec
```

```
port 513: login
```

```
port 514: shell
```

```
port 515: printer
```

```
port 6000: (X)
```

This indicates that poorsite.com is running X windows. If not protected properly (via the magic cookie or xhost mechanisms), window displays can be captured or watched, user keystrokes may be stolen, programs executed remotely, etc. Also, if the target is running X and accepts a telnet to port 6000, that can be used for a denial of service attack, as the target's windowing system will often "freeze up" for a

short period of time.

### Tip:

If you want to get some free security resources from the Internet, try this sites:

- The CERT (Computer Emergency Response Team) advisory mailing list, by sending e-mail to [cert@cert.org](mailto:cert@cert.org), and ask to be placed on their mailing list.
- The Phrack newsletter. Send an e-mail message to [phrack@well.sf.ca.us](mailto:phrack@well.sf.ca.us) and ask to be added to the list.
- The Firewalls mailing list. Send the following line in the body of the message (blank subject line) to [majordomo@greatcircle.com](mailto:majordomo@greatcircle.com): subscribe firewalls

For free software:

- COPS (Computer Oracle and Password System) is available via anonymous ftp from URL [ftp://archive.cis.ohio-state.edu, in pub/cops/1.04+](ftp://archive.cis.ohio-state.edu/pub/cops/1.04+).
- The tcp wrappers are available via anonymous ftp from URL [ftp://ftp.win.tue.nl, in pub/security](ftp://ftp.win.tue.nl/pub/security).
- Crack is available from URL <ftp://ftp.uu.net>, in [/usenet/comp.sources.misc/volume28](#).

## File Transfer Protocol (FTP)

File Transfer Protocol (FTP) is the primary method of transferring files over the Internet. By using FTP you can transfer files across the globe. Although there are sites or certain areas of a site that requires authentication, usually you can logon to a site anonymously.

Through the login process you will have to enter your username, which when connecting anonymously is the word "anonymous," and the password, which usually will be your e-mail address (not a requirement but a custom and netiquette so the sites can track the level of FTP usage). Figure 2.11 shows a login connection to an FTP site, and figure 2.12 shows the authentication process once you're connected to the site.

You must be careful with anonymous FTP, which is very similar to have someone logged on with a "guest" account on your network's server. Once a hacker is inside your FTP server, he/she can exploit it and put in danger the security of your site, so at the very least, your FTP server should always be outside of a firewall or have no connections to your server or workstations attached to your server.

## *Some of the Challenges of Using Firewalls*

One of the major challenges of security when internetworking is when protected networks, or Intranets, converge with unprotected ones, such as the Internet. This convergence forms an internetworked "blob" that can look like figure 2.13.

The scenario shown on figure 2.13 is a bit confusing, and therefore, very much prompt for security

problems. Thus, corporate IS departments rush to set in place security tools that many times are immature rather than being a complete solution to address the variety of challenges the Internet presents to the corporation's internetworking. That's why, usually what we see as a solution for the "blob" is a series of security measures that ends up looking more like a fruit salad than a controllable and efficient Internet security system. Some of the flavors we find, isolated or combined, include but are not limited to:

- Password-based security
- Customized access controls
- Encryption schemes
- Firewalls
- Proxy servers

Well, sometimes we won't find anything!

In order to take advantage of the potential the Internet and the Web brings, which includes the tremendous growth of the electronic commerce industry, businesses must be able to:

- Have an open door to the Internet, allowing the exchange of resources and corporate identity, just like a store front needs to have the other open to the public. By selling its good via a "drive-thru" environment will not last for long. Nevertheless, while open the Internet door to its internal network, the company must maintain the control over who accesses their internal resources and who doesn't.
- Identify and authenticate customers who use the Internet to access their corporate networks. This includes customers using both e-mail and tunneling connections.
- Ensure that private information sent over the public Internet can be transmitted securely for the customer's and company's protection.

These are some of the major challenges in using a firewall effectively, not as part of the blob! We, as what I would call now Internet Managers, have a challenge, which it is also the challenge of this book, to successfully and effectively address the security requirements outlined on figure 2.14, which should be accomplished by the time you finish reading it.

## Increasing Security on IP Networks

As you realized by now, network security is a broad topic that can be addressed at the data link, where packet snooping and encryption problems can occur, at the network or protocol layer, the point at which we control IP packets and routing updates, and at the application layer, where, for example, host-level bugs become issues.

As you and your organization have more and more access to the Internet, and as you expand your organization's networks, the challenge you have to provide security for your LAN/WAN and Intranets becomes increasingly difficult. You will have to determine which areas of your network you must protect, learn how to restrict your internal (and external users and customers) user access to these areas, and determine which types of network services you should filter to prevent potential security breaches.

Chapters one and this chapter addressed the many characteristics and security issues of IP protocols and

services. This chapter also identified several weaknesses at the IP protocol and services level, but also provided alternatives for increasing security on the IP networks as well as few features and processes you can use to enhance the security level of your site. These features included controls to restrict access to routers and communication servers by way of console port, Telnet, SNMP, and so on. But those measures are not enough, so you must consider implementing firewalls. Although firewall concepts were introduced on chapter one, much more on its architecture and setup needs to be discussed, but before we do that, lets take a look at few other alternatives widely and effectively used: cryptography. After all, a question remains: is it enough? That's what chapter three is all about.

[Orders](#)[Orders](#)[Back](#)[Forward](#)[Comments](#)[Comments](#)

---

[COMPUTING MCGRAW-HILL](#) | [Beta Books](#) | [Contact Us](#) | [Order Information](#) | [Online Catalog](#)

---

HTML conversions by [Mega Space](#).

This page updated on December 05, 1997 by [Webmaster](#).

[Computing McGraw-Hill](#) is an imprint of the [McGraw-Hill Professional Book Group](#).

Copyright ©1997 [The McGraw-Hill Companies, Inc.](#) All Rights Reserved.

Any use is subject to the rules stated in the [Terms of Use](#).

Orders

Orders

Back

Forward

Comments

Comments

© 1997 The McGraw-Hill Companies, Inc. All rights reserved.

Any use of this Beta Book is subject to the rules stated in the [Terms of Use](#).

## Chapter 3

# Cryptography: Is it Enough?

Never mind personal use! Encryption will be widely adopted to protect transactions over the electronic commerce industry, despite what the government concerns are with regards to national security.

The increasing growth of the electronic commerce are pushing the issue of data encryption to the main courts, as more and more there is a need for companies and netizens to protect their privacy on the Internet, as well as their commercial and financial transactions. But the government is a bit nervous about it as, for the first time, encryption can block the watchful eyes of the law enforcement agencies over individuals, which in fact, is a double-edged sword, as if powerful encryption schemes is to fall into the wrong hands it can represent freedom for crimes to be committed and go undetected.

Cryptography's main tool, the computer, is now available everywhere! Since the World War II (WWII) governments worldwide have been trying to control the use of data encryption (ask Phil Zimmermann about it!). No longer we need Colossus, the computer built during WWII to crack the German military's secret code! My 14 years old son already uses a Pentium at home, access the Internet and encrypts his files with CodeDrag!

### Note:

What about Phil Zimmermann?

He was the developer of Pretty Good Privacy, an encryption tool he placed on the Internet after he finished developing it, in which he was persecuted by the U.S. government for it. For more information and details about the whole case, check the URL <http://web.its.smu.edu/~dmcnickl/miscell/warnzimm.html>

### Tip:

What is CodeDrag?

CodeDrag, as shown on figure 3.1, is very fast encryption tool that uses a fast C-implementation of the DES-algorithm to increased speed. It was developed at the University of Linz, Austria, as an example tool to demonstrate the new possibilities of the Windows 95 shell, as CodeDrag is fully embedded into the Windows desktop. For more information you can contact the developing team at [dragon@fim.uni-linz.ac.at](mailto:dragon@fim.uni-linz.ac.at) or visit their site (and download a copy of CodeDrag) from URL <http://www.fim.uni-linz.ac.at/codeddrag/codedrag.htm>.

Since 1979, the National Security Agency (NSA) had classified any form of encryption as weapons, compared to fighter jets and nuclear missiles. However, people like Zimmermann, concerned with privacy and civil rights, have been fighting against exclusive government control of encryption. During the 70, Whitfield Diffie, of Stanford Research Institute,



developed what is today known as public key cryptography, which is discussed in more details later on this chapter.

Diffie's innovation actually created a revolution in the encryption world back there, especially among the government. The problems was that, while the government's secret agencies were still using single key schemes, which would rely upon both the sender and the receiver of an encoded message having access to the key, he proposed a dual-key approach which made it much simpler to encrypt data.

Not long time later, in 1977, a company found by three scientists from the Massachusetts Institute of Technology (MIT), RSA Data Security, introduced the first public key cryptography software and obtained US patents for the scheme.

It was in 1991 that Zimmermann, then a computer programmer, launched his "Pretty Good Privacy" (PGP) encryption software and distributed it freely on the Internet, making it internationally available. Not only his action got the governments attention to him, which lead to his persecution, but even RSA Data Security also condemn PGP, classifying it as a threat to its commercial interests.

Nowadays, even commercial software companies are developing their own encryption products. Take Netscape, for example, which developed and freely distributed their security scheme all over the Internet as well. Netscape's Secure Sockets Layer (SSL) encryption scheme uses 56 character key to increase data security. Microsoft also come up with an encryption tool, know as Private Communications Technology (PCT) protocol.

As discussed in the past two chapters, computer network security is becoming increasingly important as the number of networks increase and network size expands. Besides, the Internet has also become an extension of the protected networks of a corporation. Until last year, Intranets were something new, but only a little more than an year later we are already talking and investing into Extranets. As the sharing of resources and information worldwide (Cyberspace included!) becomes easier, the ability to protect information and resources against unauthorized use becomes critical.

By now you already realized that it is not possible to have a 100% secure network. At the same time, information needs to be accessible to be useful. Balancing accessibility and security is always a tradeoff and is a policy decision made by management.

Good security involves careful planning of a security policy, which should include access control and authentication mechanisms. These security strategies and procedures can range from a very simple password policy to complex encryption schemes. Assuming that you have already implemented at least a password policy at your organization (you did it right?!), this chapter will be discussing about the many levels and types of encryption schemes and when it is enough. Is it?

## ***Introduction***

Encrypting the information of your company can be an important security method and provides one of the most basic security services in a network: authentication exchange. Other methods, such as Digital Signatures and data confidentiality, also use encryption.

## ***Symmetric Key Encryption (Private Keys)***

There are several encryption techniques available on the market, using several kinds of algorithms, but the two main ones are the ones using keys and those not relying on keys at all.

Encryption techniques not using any keys are very simple and they work by transforming, scrambling, the information being encrypted. For instance, you could encrypt a message written in English text by just adding a number to the ASCII value of each letter, which could give a result as shown on figure 3.2. Although apparently secure, this sort of algorithm is not so secure. Actually, they are very easy to decipher. Once you learn the algorithm you will be able to decipher the encrypted information.

There are more secure algorithms that use a sort of key along with the data. Two major types of encryption algorithms are private key encryption and public key encryption, to be discussed in more details later. A private key is also called a single key, secrete key, or symmetric key. A public key is also called an asymmetric key.

With private key encryption algorithms, only one key exists. The same key value is used for both encryption and decryption. In order to ensure security, you must protect this key and only you should know it. Kerberos, for example, which is discussed in more details later on this chapter, is an authentication protocol that uses private key algorithms.

Another characteristic of private key encryption is that the keys used are usually small, making its algorithms computation relatively fast and easier than asynchronous ones.

One of the main limitations of using private key encryption is when distributing it to everyone who needs it, especially because the distribution itself must be secure. Otherwise you could expose and compromise the key and therefore, all the information encrypted with it. Thus, it becomes necessary for you to change your private key encryption every so often.

If you only have private key schemes available to you, I recommend you to use it with digital signatures, as they are much more versatile and secure.

## ***Data Encryption Standard (DES)***

The Data Encryption Standard (DES) is one of the most commonly used private key algorithm. DES was developed by IBM and became a U.S. Government standard in 1976. This is a well known algorithm, with a large implementation base in commercial and government applications. As mentioned earlier, Kerberos uses the DES algorithm to encrypt messages and create the private keys used during various transactions.

DES is very fast. According to RSA Labs, when DES is implemented entirely in software, it is at least 100 times faster than the RSA algorithm. But if implemented in hardware, DES can outperform the RSA algorithm by 1000 or even 10000 times since DES uses S-boxes, which have very simple table-lookup functions, while RSA depends on very-large-integer arithmetic.

DES uses the same algorithm for encryption and decryption. The key can be just about any 64-bit number. Because of the way the algorithm works, the effective length is 56 bits. NIST certified DES for use as an official US Government encryption standard but only for "less-than-top-secret secret material." Although DES is considered very secure, there are actually two known ways to break it:

- Through an exhaustive search of the keyspace, providing a total of  $2^{56}$  (about  $7.2 \cdot 10^{16}$ ) possible keys, which would take about 2,000 years if you were to test one million keys every second, and
- Good luck!

Until recently, DES was never been broken and was believed to be secure. But a group of Internet users, working together in a coordinated effort to solve the RSA DES challenge, see figure 3.3, for over four months finally broke the algorithm. The group checked nearly 18 quadrillion keys, finding the one correct key to reveal the encrypted message:

**"Strong cryptography makes the world a safer place."**

### **Note:**

The U.S. Government forbids export of hardware and software products that contain certain DES implementations. American exporters must adhere to this policy even though implementations of DES are widely available outside of the United States.

The group used a technique called "brute-force", where the computers participating in the challenge began trying every possible decryption key. There are over 72 quadrillion keys (72,057,594,037,927,936). At the time the winning key was reported to RSA Data Security, Inc, in June of 97, the group, known as DESCHALL (DES Challenge), had already searched almost 25% of the total possibilities. During the peak time of the group's efforts 7 billion keys were being tested per second. Figure 3.4. is a screenshot of the DESCHALL site, located at URL <http://www.frii.com/~rcv/deschall.htm>

Although DES was cracked, it has remained a secure algorithm for over 20 years. The brute-force attack used against DES is very common when trying to decipher an algorithm. Although you must try all the possible  $2^{56}$  keys of DES on a

plaintext and match the result against the known corresponding ciphertext, by using differential cryptanalysis you could reduce the amount of tryouts to  $2^{47}$ , which is still a big project to undertake. If DES were to use a key longer than 56-bit key, the possibilities of cracking it would be nearly to impossible.

## International Data Encryption Algorithm (IDEA)

International Data Encryption Algorithm (IDEA) is one of the best and most secure algorithms available. Developed by Xuejia Lai and James Massey of the Swiss Federal Institute of Technology, IDEA uses a block size of 64 bits, sufficiently strong against cryptanalysis. IDEA also uses a cipher feedback operation that strengthens the algorithm even further. In this mode, ciphertext is used as input into the encryption algorithm.

Another important feature of IDEA is its key length of 128 bits. As you saw with DES, the longer the key, the better. Also, IDEA gives no clues to the contents of the plain-text when you try to decipher it, it spreads out a single plain-text bit over many ciphertext bits, hiding the statistical structure of the plain-text completely.

Nevertheless, IDEA does have minimum requirements, and it will need a 64 bits of message text in a single coding block in order to ensure a strong ciphertext. If you're encrypting large amount of data, it shouldn't be a problem, but not indicated for situations where 1 byte keystrokes are exchanged. Clearly, IDEA is ideal for FTP, when large amount of data are transmitted. However, as you might guess, it would work very poorly with Telnet.

Fauzan Mirza developed a secure file encryption program called Tiny IDEA (<http://www.dcs.rhnc.ac.uk/~fauzan/tinyidea.html>). Figure 3.5 shows a screenshot of Tiny IDEA's site, where the program can be downloaded and instructions and additional information about the program is available.

## CAST

Developed by Carlisle Adams and Stafford Tavares, CAST algorithm uses a 64-bit block size and a 64-bit key. The algorithm uses a six S-boxes with an 8-bit input and a 32-bit output. Don't even ask me about the constitution of these S-boxes, as it is very complicated and out of the scope of this book. For that I strongly recommend Bruce Schneier's book "Applied Cryptography," by John Wiley (ISBN 0-471-11709-9), which is a great book for those wanting to dig into cryptography.

CAST encryption is done by dividing the plaintext block into two smaller blocks, left and right blocks. The algorithm has eight rounds and in each round one half of the plaintext block is combined with some key material using a function "f" and then XORed with the other block, the left one to form a new right block. The old right hand becomes the new left hand. After doing this eight times the two halves now will be concatenated as a ciphertext. Table 3.1 shows the "f" function, according to the example of Schneier in the above mentioned book, page 335, which is very simple.

**Table 3.1 - The Function used by CAST for encryption of plaintext blocks into a ciphertext.**

|   |                                                                                                                               |
|---|-------------------------------------------------------------------------------------------------------------------------------|
| 1 | Divide the 32-bit input into four 8-bit quarters: a, b, c, d.                                                                 |
| 2 | Divide the 16-bit subkey into two 8-bit halves: e, f.                                                                         |
| 3 | Process a through S-box 1, b through S-box 2, c through S-box 3, d through S-box 4, e through S-box 5, and f through S-box 6. |
| 4 | XOR the six S-box outputs together to get the final 32-bit output.                                                            |

**Note:**

What are S-boxes?

S-boxes, or selection boxes, are a set of highly non-linear functions, which are implemented in DES as a set of lookup tables. They are the functions that actually carry out the encryption and decryption processes under DES.

Figure 3.6 is a screenshot of a DES S-boxes site at the College of William and Mary, courtesy of Serge Hallyn at URL <http://www.cs.wm.edu/~hallyn/des/sbox.html>, which is worthwhile for you to check. Also, for your convenience, figures 3.7 through 3.14 are screenshots of DES S-box 1 through 8 respectively.

## Skipjack

Skipjack is an encryption algorithm developed by the National Security Agency (NSA) for the Clipper chips. Unfortunately, not much is known about the algorithm, as it is classified as secret by the US government. It is known that this is a symmetric algorithm, which uses a 80-bit key and has 32 rounds of processing per each encrypt or decrypt operation.

The Clipper-chip is a commercial chip made by NSA for encryption, using the Skipjack algorithm. AT&T does have plans to be using the Clipper for encrypted voice phone lines.

### But is Skipjack Secure?

As far as I know, NSA has been using Skipjack to encrypt it's own messaging system, so that leads to think the algorithm itself is secure. Skipjack uses 80-bit keys, which means there are  $2^{80}$  (approximately  $10^{24}$ ) or more than 1 trillion trillion possible keys to be used!! This means that (ready for this?!) it would take more than 400 billion years for every key of the algorithm to be tried!

To give you a better perspective, if we were to assume the use of 100,000 RISC computers, each with the capability of cranking about 100,000 encryptions per second, it would still take about 4 million years for a code to be broken.

The developers of Skipjack estimated that the cost of processing power to break the algorithm is halved every eighteen months, and based on that that it would take at least 36 years before the cost of breaking Skipjack by brute-force to be equal to the cost of breaking DES today. Thus, they believe that there is no risk for Skipjack to be broken within the next 30-40 years. Besides, it is also known that the strength of Skipjack against a cryptanalytic attack does not depend on the secrecy of the algorithm, so even if the algorithm were to be known, Skipjack would still be believed to be very secure.

**Tip:**

For detailed information on Skipjack, check the URL [http://www.cpsr.org/cpsr/privacy/crypto/clipper/skipjack\\_interim\\_review.txt](http://www.cpsr.org/cpsr/privacy/crypto/clipper/skipjack_interim_review.txt), which provides a complete overview about it.

Clipper uses Skipjack with two keys, and whoever knows the chip's "master key" should be able to decrypt all messages encrypted with it. Thus, NSA could, at least in thesis, decrypt Clipper-encrypted messages with this "master-key" if necessary. This method of tampering with the algorithms is what is so called and known as the key escrow.

There are many resistance from concerned citizens and the business sector against the Clipper-chip as they perceive it as an invasion of their privacy. If you check the URL <http://www.austinlinks.com/Crypto/non-tech.html> you will find detailed information about the Clipper wiretap chip.

## RC2/RC4

RC4, which used to be a trade secret until the source code was posted in the USENET, is a very fast algorithm, designed by RSA Data Security, Inc. RC4 is considered a strong cipher, but the exportable version of Netscape's Secure Socket Layer (SSL), which uses RC4-40, was recently broken by at least two independent groups which took them about eight days.

Table 3.2 gives you an idea of how the different symmetric cryptosystems compare to each other.

**Table 3.2 - A Symmetric Cryptosystems Comparison Table**

| Cipher       | Security   | Speed (486 pc) | Key length |
|--------------|------------|----------------|------------|
| DES          | low        | 400 kb/s       | 56 bits    |
| 3DES         | good       | 150 kb/s       | 112 bits   |
| IDEA         | good*      | 200 kb/s       | 128 bits   |
| 3IDEA        | very good* | ~100 kb/s      | 256 bits   |
| Skipjack     | good*      | ~400 kb/s      | 80 bits    |
| CLIPPER chip | good**     | -              | 80 bits    |

\* the algorithm is believed to be strong

\*\* the algorithm itself is good, but it has a built-in weakness

## Asymmetric Key Encryption/Public Key Encryption:

In this cryptosystem model, two keys, used together, are needed. One of the keys always remains secret while the other one becomes public. You can use each key for both encryption and decryption. Public key encryption helps solve the problem of distributing the key to users.

Some examples of public key encryption usage includes:

- **Certificates** to ensure that the correct public and private keys are being used in the transaction.
- **Digital Signatures** to provide a way for the receiver to confirm that the message came from the stated sender. In this case, only the user knows the private key and keeps it secret. The user's public key is then publicly exposed so that anyone communicating with the user can use it.
- **Plaintext** encrypted with a private key can be deciphered with the corresponding public key or even the same private key.

One of the main public key encryption algorithm is RSA, which was named after its inventors, Rivest, Shamir, and Adleman. These public key algorithms always have advantages and disadvantages. Usually, the encryption and decryption of the algorithms use large keys, often with 100 or more digits. That's why the industry has the tendency to resolve key management and computing overhead problems by using smart cards such as SecureID and so on.

Zimmermann's Pretty Good Privacy (PGP), is an example of a public-key system, which is actually becoming very popular for transmitting information via the Internet. These keys are simple to use and offer a great level of security. The only inconvenient is to know the recipients' public key, and as its usage increases, there are a lot of public keys out there, without a central place to be stored. But there is a "global registry of public keys" effort at works, as one of the promises of the new LDAP technology.

**Note:****What about LDAP?**

LDAP is an acronym for *Lightweight Directory Access Protocol*, which is a set of protocols for accessing information directories. Based on the X.500 protocol, LDAP is much simpler to use and supports TCP/IP (X.500 doesn't), necessary for any type of Internet access.

With LDAP an user should be able to eventually obtain directory information from any computer attached to the Internet, regardless of the computer's hardware and software platform, therefore allowing for an specific address or public-keys to be found without the need for clearing house sites such as Four11 (<http://www.four11.com>) or similar.

## RSA

RSA, developed invented in 1977 by Ron Rivest, Adi Shamir, and Leonard Adleman (RSA), is a public-key cryptosystem for both encryption and authentication. RSA has become a sort of standard as it is the most widely used public-key cryptosystem.

RSA works as follows: take two large primes,  $p$  and  $q$ , and find their product  $n = pq$ . Choose a number,  $e$ , less than  $n$  and relatively prime to  $(p-1)(q-1)$ , and find its inverse,  $d$ , mod  $(p-1)(q-1)$ , which means that  $ed = 1 \text{ mod } (p-1)(q-1)$ ;  $e$  and  $d$  are called the public and private exponents, respectively. The public key is the pair  $(n,e)$ ; the private key is  $d$ . The factors  $p$  and  $q$  must be kept secret, or destroyed.

It is difficult (presumably) to obtain the private key  $d$  from the public key  $(n,e)$ . If one could factor  $n$  into  $p$  and  $q$ , however, then one could obtain the private key  $d$ . Thus the entire security of RSA is predicated on the assumption that factoring is difficult; an easy factoring method would break RSA.

RSA is fast, but not as DES. The fastest current RSA chip has a throughput greater than 600 Kbits per second with a 512-bit modulus, implying that it performs over 1000 RSA private-key operations per second.

### Is RSA Algorithm Secure?

The security of RSA will depend on the length of the keys used. A 384 bits key can be broken much easier then a 512 bits, which is still probably insecure and breakable. But if use a 768 bits key, then the amount of possible combinations grow substantially. According to RSA's FAQ (<http://www.rsa.com/rsalabs/newfaq/secprserv.htm>), as seen on figure 3.15, a 1024 bits key should be secure for decades.

But this doesn't mean that RSA is unbreakable. You you can compute  $e$ -th roots mod  $n$  you can break the code. Since  $c = m^e$ , the  $e$ -th root of  $c$  is the message  $m$ . This attack would allow someone to recover encrypted messages and forge signatures even without knowing the private key.

Also, according to RSA's FAQ at the URL above, the cryptosystem is very vulnerable to chosen-plaintext attacks, and a good guess can reveal the used key. Thus, it is advisable to include some random data (at least 64 bits) to the encrypted plaintext.

## Digital Signature Standard (DSS)

Digital Signature Standard (DSS) is a US government standard for digital signaturing. DSS has some problems being the leakage of secret data one of them. Also, if you use the same random number twice when generating the signature, the secret key will be revealed. Further, with Diffie-Hellman and RSA cryptosystem methods being available, which are much better then DSS, I see no reason for using DSS.

Table 3.3 shows a comparison table of the asymmetric cryptosystems available.

**Table 3.3 - Asymmetric Cryptosystems Comparison Table**

| Cipher                | Security | Speed | Key length        |
|-----------------------|----------|-------|-------------------|
| <b>RSA</b>            | good     | fast  | varies(1024 safe) |
| <b>Diffie-Hellman</b> | good     | < RSA | varies(1028 safe) |
| <b>DSS</b>            | low      | -     | 512 bits          |

Figure 3.16 shows a summary overview of how public/private keys are generated.

## Message Digest Algorithms

Message Digest (MD) algorithm are developed to take any message as input and produce an output of 128-bit "message digest," also called "fingerprint." Two messages can never have the same message digest. There are three versions of message digest available, the MD2, MD4 and MD5, which is discussed in more details next.

### MD2, MD4 and MD5

The Message Digest Algorithm 5 (MD5) is the latest version of the MDs, a secure hash algorithm, which was developed by RSA Data Security, Inc. MD5 can be used to hash an arbitrary length byte string into a 128 bit value, as its earlier versions. However, MD5 is considered a more secure hash algorithm and it is widely in use.

MD5 processes the input text in 512-bit blocks, divided into 16 32-bit sub-blocks. The output is a set of 4 32-bit blocks, which are concatenated to a single 128-bit hash value.

Although very secure, MD5 was recently reported having some potential weaknesses in it, which are breakable in some cases. It is also said that one could build a special-purpose machine costing a few million dollars to find a plaintext matching a given hash value in a few weeks, but it can be easier than that.

For instance, Microsoft Windows NT uses MD4, which is discussed on the next section, to encrypt the password entries that are stored in its Security Account Manager (SAM) database. Earlier this year, around the Spring of 1997, a weakness on the security of Windows NT was exploited, which involved the security of the MD4 as well.

Couple utilities widely available on the Internet, called PWDUMP (you can download it from <http://www.masteringcomputers.com/util/nt/pwdump.htm>) and NTCRACK (also downloadable from <http://www.masteringcomputers.com/util/nt/ntcrack.htm>), were used to crack users passwords on NT. The SAM database, target of PWDUMP, is the one responsible for storing the passwords on NT. But SAM doesn't really store the passwords in plaintext, but a hash value of it, as shown on figure 3.17.

If you carefully check figure 3.17 you will find out that the hash of my password on my computer is exposed, but the password is still UNKNOWN. When a password is entered for the first time on NT, the system uses MD4 to generate a hash of that password, which is exposed by PWDUMP, as shown in the fourth line, in front of the field "NTHASH." This hash is then encrypted before it is stored in the SAM database.

The problem here is that PWDUMP is capable of finding out the function used to encrypt the values of this hash created by MD4. Since the encrypting process of MD4 is know (remember that earlier in this chapter we mentioned that the source code of MD4 was posted on the USENET?), the password can be found by a reverse engineering process. You can then use NTCRACK, as well as many other tools derived from it, to feed MD4's encryption system with a list of words (from a dictionary, for example) and compare the value of the hashes of each word until you find the one that matches the password, which is easier on NT since it doesn't use a randomic elements (SALT) during the encryption process—this doesn't mean that UNIX systems are more secure, as they use SALT, it just would delay the decryption process a little longer!

In order to exploit NT's password encryption system and MD4 is not a big deal here. The major challenge is that you will need to connect to the machine you want to exploit as an administrator. Once done that, here it is what you'll need to do:

1. Create a temporary directory where you will run the tools and make sure both, PWDUMP and NTCRACK reside there.
2. Type PWDUMP > LIST.TXT (or any other suggestive name you want. This file will store all the password hashes PWDUMP will find).
3. Now it is time to use NTCRACK! Type NTCRACK PASSWORDS LIST.TXT > CRACKED.TXT. (PASSWORDS is the name of the file containing words, preferably a whole dictionary, in ASCII format. NTCRACK comes with a basic dictionary file, you should add more words to it. Ask your secretary to enter the whole Webster there! Once the process is finished you just need to open the file named CRACKED.TXT with any text editor and check which passwords were cracked.

The NTCRACK version listed earlier is one of the most updated one at the time this chapter is written, mid-June of 1997. This version not only checks the passwords against its basic dictionary, but also checks for passwords that are identical to the username, which I used as an example for a cracked password on figure 3.18. Note that only passwords part of the dictionary file are cracked. That's why it's so important to use long passwords, eight characters or more and not found in any dictionary.

If you want to try this cracking tool on yourself, you can try it out on the Web. All you will need is to be running Internet Explorer, which also exposes its security flaws, and access the URL <http://www.efsl.com/security/ntie/>. There, click on the hyperlink "TRY IT." The system should provide an output with your password exposed, as shown on figure 3.18, if your password was part of its dictionary file!

As you can see on figure 3.8, whereas on the previous figure the password was unknown, now it lists my last name, GONCALVES, as it checked for passwords identical to account name.

You should know that MD5 is considered to be relatively more secure than MD4 and good enough for most purposes.

## ***Secure Hash Standard/Secure Hash Algorithm (SHS/SHA)***

Secure Hash Algorithm (SHA), also known as Secure Hash Standard (SHS) was developed by the US Government and has the ability to produce a 160-bit hash value from an arbitrary length string.

SHS is structurally similar to MD4 and MD5, only about 25% slower than MD5 but as a trade-off much more secure, because it produces message digests that are 25% longer than those produced by the MD functions, which makes it much more secure to brute force attacks than MD5.

## **Certificates**

To guarantee the authenticity of users and their keys, the Public key system requires a third party who is trusted by, and independent of, all the other parties communicating with each other.

This third party is called the Certification Authority (CA), because it is their job to certify that the owner of a Public key really is who they claim to be. To certify a Public key, the CA (such as VeriSign) creates a certificate that consists of some of the user's identification details and the user's Public key. The CA then digitally signs this certificate with their own Private key to create a Public Key Certificate.

Users can check the authenticity of another user's Public key by verifying the CA signature on the certificate using the CA's Public key, which is made widely available to the public.

After decrypting the message, the receiver verifies the sender's digital signature. To do this, a digest of the document is created using the same hash algorithm that created the original signature. At the same time, the digital signature that was attached to the document is decrypted using the sender's Public key. This creates a digest of the digital signature.

The digests of the document and the digital signature are then compared. If there is even the slightest difference between the



two, the signature is rejected. If the digests match exactly, the receiver knows that the document was not changed in transit, and can be sure of the identity of the sender.

Since the sender is the only person who has access to the Private key used to sign the message, they can't deny having sent it. Figure 3.19 shows a process where a digital signature is verified.

## Certificate Servers

Certificate Servers are applications developed for creating, signing, and managing standard-based, public-key certificates. Organizations use Certificate Servers, such as Netscape's certificate server ([http://home.netscape.com/comprod/server\\_central/support/faq/certificate\\_faq.html#1](http://home.netscape.com/comprod/server_central/support/faq/certificate_faq.html#1)) to manage their own public-key certificate infrastructure rather than relying on an external Certificate Authority service such as VeriSign, as discussed in the previous section.

Another vendor, OpenSoft (<http://www.opensoft.com/products/expressmail/overview/certserver/>) also provides Certificate Server technology for Windows NT and Windows 95 platforms. OpenSoft, uses an architecture based on the new Distributed Certificate System (DCS), which makes it a reliable public key distribution system. Figure 3.20 is a screenshot of OpenSoft's Certificate Server page.

### Note:

What about DCS?

The DCS server is a speed-optimized certificate server, based upon the DNS model. The server initially only supports four resource record types: certificate records (CRT), certificate revocation lists (CRL), certificate server records by distinguished name (CS), and certificate server records by mail domain (CSM).

As the Distributed Certificate System is intentionally extensible, new data types and experimental behavior should always be expected in parts of the system beyond the official protocol. As in DNS, the DCS server uses a delimited, text-based file format named the DCS master files. The DCS server allows multiple master-files to be used in conjunction, as well as a 'root' file, where authoritative root server information is stored.

For more information on DCS, check OpenSoft's Web Site at URL <http://www.opensoft.com/dcs/>. The following section, "DCS: What is Under the Hood?," is an edited (stripped) version of the full document available at OpenSoft's URL listed above, which holds the copyrights of this document.

## DCS: What is Under the Hood?

As briefly discussed in the previous section, the Distributed Certificate System (DCS) server is a speed-optimized certificate server, based upon the DNS model. The server initially only supports four resource record types:

- certificate records (CRT),
- certificate revocation lists (CRL),
- certificate server records by distinguished name (CS), and
- certificate server records by mail domain (CSM).

As the DCS is intentionally extensible, new data types and experimental behavior should always be expected in parts of the system beyond the official protocol. As in DNS, the DCS server uses a delimited, text-based file format named the DCS master files. The DCS server allows multiple master-files to be used in conjunction, as well as a 'root' file, where authoritative root server information is stored.

# The Certificate Server

A certificate server allows a user agent or other certificate servers to query for certificate information. The following is a brief overview of the characteristics of a certificate server:

1. A certificate server maintains the following records:

- A *CRT record* has three fields:
  - distinguished name,
  - record type (CRT),
  - the certificate
- A *CRL record* has three fields:
  - CA's distinguished name,
  - record type (CRL),
  - the signed CRL
- A CS record has three fields:
  - distinguished name segment,
  - record type (CS),
  - server address
- A CSM record has three fields:
  - domain name,
  - record type (CSM),
  - server address

2. In a CRT or CRL query, a user agent sends a request for a certificate or CRL to a certificate server, given a distinguished name:

- if a CRT or CRL record is not present, the server searches for a CS record to see where the certificate may be found, otherwise the server asks a DCS root server where to look for this certificate or CRL
- if the CRT or CRL record is present, the certificate or CRL is returned

3. In a CS(M) query, a distinguished name segment may be an attribute, or set of attributes:

- Refer to RFC 1779 ("A String Representation of Distinguished Names") to obtain the necessary format for distinguished names in CS(M) queries.
- At the user agent, marking an attribute or set of attributes in the distinguished name allows the server to decide how to look for the corresponding certificate on another server via a CS query
- Only the marked attribute or set of attributes is used in a CS query, this marked set is the common element in distinguished names of certificates located at the server with the correct key, but not all certificates at this location have this common element
- This query method is similar to how DNS uses the NS record to find the address of servers with a common domain
- By default, a user agent uses the e-mail attribute as the marked attribute, if no other attribute or set of attributes is marked. From the e-mail address, the domain name is extracted and then used in a CSM query. If there is no email attribute and no other marked attribute, then the first attribute in the first set is used as the marked attribute.
- A user agent may also request CRLs from the DCS in the above manner.

4. CRT, CRL, and CS(M) records are stored in a DCS master file which is similar to the DNS master file format.

## DCS Topology

A common topology of multiple DCS hosts and their role in the Internet is represented on figure 3.21.

On figure 3.21, note that:

1. Edit DCS master files . Records used: CRT, CRL, CS, CSM
2. Request to the Certificate Authority for CRL(s). Records used: CRL
3. Request to the certificate server for certificates and CRLs. Record used: CRT, CRL
4. DCS inter-server communication. Records used: CS, CSM

The DCS topology illustrates the high-speed nature of this system. A user agent may query a local certificate server and in milliseconds receive a transmission of the desired certificate or CRL from that certificate server or perhaps another server located anywhere on the Internet.

## DCS Protocol

Refer to RFCs 1032-1035 on the DNS protocol for the exact syntax on DCS queries. The DCS query protocol will have the same format as the DNS query protocol. The syntax of distinguished names within DCS queries will conform to RFC 1779 ("A String Representation of Distinguished Names").

All communication inside of the DCS protocols are carried in a single format called a DCS message (DCSM). The top level format of message is divided into 5 sections, just like with DNS, some of which are empty in certain cases, as shown on figure 3.22.

Looking at figure 3.22, the header section is always present. The header includes fields that specify which of the remaining sections are present, and also specify whether the message is a query or a response, a standard query or some other opcode, etc.

The names of the sections after the header are derived from their use in standard queries. The question section contains fields that describe a question to a name server. These fields are a query type (as the QTYPE in DNS), a query class (as the QCLASS in DNS). The last three sections have the same format: a possibly empty list of concatenated DCS records. The answer section contains RRs that answer the question; the authority section contains RRs that point toward an authoritative name server; the additional records section is not used in the DCS.

## Header Section Format

The header contains the following fields, as shown on figure 3.23:

- **ID** - A 16 bit identifier assigned by the program that generates any kind of query. This identifier is copied the corresponding reply and can be used by the requester to match up replies to outstanding queries.
- **QR** - A one bit field that specifies whether this message is a query (0), or a response (1).
- **OPCODE** - A four bit field that specifies kind of query in this message. This value is set by the originator of a query and copied into the response. The values are:
  - **0** - a standard query (QUERY)
  - **1** - an inverse query (IQUERY) (the DCS does not support it)
  - **2** - a server status request (STATUS)
  - **3** - a simple query. The certificate server makes a search of an information until finds a first required DCS record (SMQUERY).
  - **4** - an update query. A CA sets this type when sending to a certificate server new certificates or a CRL(UQUERY).
  - **5-15** - reserved for future use (in DCS)
- **AA** - Authoritative Answer - this bit is valid in responses, and specifies that the responding name server is an authority for the distinguished name in question section. Note that the contents of the answer section may have multiple owner names because of aliases. The AA bit corresponds to the name which matches the query name, or the

first owner name in the answer section.

- **TC** - Truncation - specifies that this message was truncated due to length greater than that permitted on the transmission channel.
- **RD** - Recursion Desired - this bit may be set in a query and is copied into the response. If RD is set, it directs the name server to pursue the query recursively. Recursive query support is optional.
- **RA** - Recursion Available - this bit is set or cleared in a response, and denotes whether recursive query support is available in the name server.
- **Z** - Reserved for future use. Must be zero in all queries and responses.
- **RCODE** - Response code - this 4 bit field is set as part of responses. The values have the following interpretation:
  - **0** - No error condition
  - **1** - Format error - The certificate server was unable to interpret the query.
  - **2** - Server failure - The DCS server was unable to process this query due to a problem with the certificate server.
  - **3** - Name Error - Meaningful only for responses from an authoritative name server, this code signifies that the distinguished name referenced in the query does not exist.
  - **4** - Not Implemented - The certificate server does not support the requested kind of query.
  - **5** - Refused - The certificate server refuses to perform the specified operation for policy reasons. For example, a certificate server may not wish to provide the information to the particular requester, or a certificate server may not wish to perform a particular operation, such as zone transfer, for particular data.
  - **6-15** - Reserved for future use.
- **QDCOUNT** - an unsigned 16 bit integer specifying the number of entries in the question section.
- **ANCOUNT** - an unsigned 16 bit integer specifying the number of RRs in the answer section.
- **NSCOUNT** - an unsigned 16 bit integer specifying the number of RRs in the authority records section.
- **ARCOUNT** - an unsigned 16 bit integer specifying the number of resource records in the additional records section. In the DCS protocol this value must be 0.

## Question Section Format

The question section is used to carry the "question" in most queries, such as the parameters that define what is being asked. The section contains QDCOUNT (usually 1) entries, each of the following format, as showing on figure 3.24:

where:

- **QNAME** - a DER encoded distinguished name.
- **QTYPE** - a two octet code which specifies the type of the query. The values for this field include all codes valid for a TYPE field.
- **QCLASS** - a two octet code that specifies the class of the query. This field is used for compatibility with the DNS only. For DCS it must equal the IN (the Internet).

## The DCS Record

The answer and authority all share the same format: a variable number of resource records, where the number of records is specified in the corresponding count field in the header.

Each resource record has the following format, as shown on figure 3.25:

where:

- **NAME** - A DER encoded distinguished name. If its first attribute is the e-mail address, the server finds an information by e-mail address. In another case it finds by whole distinguished name. Maybe in the query a distinguished name attribute may contain a star symbol ('\*') as wildcard instead a value. Then any value of this attribute will satisfy that template. In fact, if value equals an star symbol, then the server checks only an existence of this attribute and ignores its value.

- **TYPE** - Two octets containing one of the DCS record types. This field specifies the meaning of the data in the RDATA field.

for CS record The Type value is 1001

for CSM record The Type value is 1002

for SOC record The Type value is 1003

for SOCM record The Type value is 1004

for CRT record The Type value is 1005

for CRL record The Type value is 1006

- **AXFR** - 252 A request for a transfer of entry zone (it is identical to the DNS query). This value is same the DNS AXFR.
- **CLASS** - two octets which specify the class of the data in the RDATA field. For the DCS this value must be equal the IN.
- **TTL** - a 32 bit unsigned integer that specifies the time interval (in seconds) that the resource record may be cached before it should be discarded. Zero values are interpreted to mean that the RR can only be used for the transaction in progress, and should not be cached. The each DCS record contains a time value. This field may not be necessary.
- **RDLENGTH** - an unsigned 16 bit integer that specifies the length in octets of the RDATA field. In DCS the DATA is the DER encoded value. Thus the RDATA contains its length. Therefore this field is not used.
- **RDATA** - A DER encoded ASN.1 type. The format of this information varies according to the TYPE of the RR.

If you would like to have more information about DCS message compression and transport, as well as server algorithm, please check OpenSoft URL at URL <http://www.opensoft.com/dcs/>, as I feel that this kind of information already goes beyond the scope of this book.

## Key Management

The only reasonable ways to protect the integrity and privacy of information is to rely upon the use of secret information in the form of private keys for signing and/or encryption, as discussed earlier in this chapter. The management and handling of these pieces of secret information is generally referred to as "key management." This includes the process of selection, exchange, storage, certification, expiration, revocation, changing, and transmission of keys. Thus, most of the work in managing information security systems lies in the key management.

The use key management within public key cryptography, as seeing earlier, is appealing because it simplifies some of the problems involved in the distribution of secret keys. When a person sends a message, only the receiver can read it. This without having any need for the receiver to know the original key used by the sender or agree on a common key, as the key used for encryption is different from the key used for decryption.

Key management not only provides convenience for encrypted message exchange, but also provides the means to implement digital signatures. The separation of public and private keys is exactly what is required to allow users to sign their data, allow others to verify their signatures with the public key, but not have to disclose their secret key in the process.

## Kerberos

The Kerberos protocol provides network security by regulating user access to networking services. In a Kerberos environment, at least one system runs the Kerberos Server. This system must be kept secure. The Kerberos Server, referred to as a trusted server, provides authentication services to prove that the requesting user is genuine. Another name for the Kerberos Server is the Key Distribution Center (KDC).

Other servers on the network, and all clients, are assumed by the system administrator to be untrustworthy. For the Kerberos protocol to work, all systems relying on the protocol must trust only the Kerberos server itself.

In addition to providing authentication, Kerberos can supply other security services such as:

- Data integrity
- Data confidentiality

Kerberos uses private key encryption based on the Data Encryption Standard (DES). Each client and server has a private DES key. The Kerberos protocol refers to these clients and servers as principals. The client's password maps to the client's private key.

### Tip:

For a great source of information on Kerberos and its applicability in the network security environment, check Process Software Web site at <http://www.process.com>. Not only they are one of the leading TCP/IP (including IPv6!) solution company, but also have a vast resource of information on IPv6, Kerberos and TCP/IP.

The Kerberos Server maintains a secure database list of the names and private keys of all clients and servers that are allowed to use the Kerberos Server's services. Kerberos assumes that all users (clients and servers) keep their passwords secure.

The Kerberos protocol solves the problem of how a server can be sure of a client's identity. Kerberos does this by having both the client and server trust a third party, in this case, the Kerberos Server. The Kerberos Server verifies the client's identity.

## Getting to Know Kerberos Terms

Some of the terms commonly associated with Kerberos include:

- **Principal** - Kerberos refers to clients and servers as principals and assigns each one a name. An example of the general naming format is `name.instance@realm`.

Kerberos refers to clients and servers as principals and assigns each one a name. An example of the general naming format is `name.instance@realm`.

- **name** — For clients, this is the user's login name; for servers, it is the name of the service provided, usually `rcmd`.
- **instance** — This is usually omitted and unnecessary for clients; for Kerberos administrators, the value is `admin`; for servers, it identifies the machine name of the application server that has Kerberos authentication support. For example, if the `rlogin` server on `hostX` has Kerberos authentication support, the principal would have the following format: `rcmd.hostX@your_realm`
- **realm** — Is associated with all principals in a Kerberos database and is the name of a group of machines, such as those on a LAN; it identifies the Kerberos domain.

You can omit the instance and realm components from some principals. For example, a possible principal for `joshua` (for user Joshua in the local domain) could be `joshua@xuxu.com` for user Jones in the `xuxu.com` domain. A possible principal could also be `rcmd.hostX` (for the `rlogin` server in the local domain) or `rcmd.hostX@xuxu.com` (for the `rlogin` server on `hostX` in the domain `xuxu.com`).

- **Ticket-granting ticket** - A ticket-granting ticket contains an encrypted form of the user's Kerberos password. Use it to obtain application service tickets from the Kerberos server. You cannot use Kerberos authentication without first having this ticket-granting ticket.

The ticket-granting ticket has an associated lifetime that the Kerberos server specifies. This lifetime is generally eight hours. You can use the same ticket over and over again, until you no longer need the ticket or it expires.

- **Service ticket** - Kerberos uses service tickets to verify a client's identity to an application server. The Kerberos server encrypts the service ticket with the application server's private key. Only that application server can decrypt

the service ticket.

- **Authenticator** - The Kerberos protocol uses authenticators to prevent eavesdroppers from stealing a ticket. The client sends a new authenticator with each service request. An authenticator consists of the client's name and IP address, and a timestamp showing the current time.

The server uses the information in the authenticator to confirm that the rightful owner presents the accompanying ticket. For this to be true, the client and server must synchronize their clocks. One way of doing this is through the Network Time Protocol (NTP).

## What is in a Kerberos Session

The Kerberos protocol is an authentication system for open systems and networks. Kerberos uses a set of encrypted keys and tickets for authentication, making authentication between two systems secure.

Standard authentication methods, on the other hand, are not secure because the username and password are generally sent across the network in clear, readable text.

## A Typical Kerberos Session

The following describes the general sequence of a Kerberos session, as shown on figure 3.2 6):

1. The Client submits a request to the Kerberos Server to obtain a ticket-granting ticket (TGT). The Kerberos Server consults the Kerberos database (KDB) to get the user's Kerberos password, and then encrypts it.
2. The Kerberos Server sends the encrypted password in the TGT to the Client. When the Client receives the TGT, it requests the user's Kerberos password, then encrypts the password and compares it to the password in the TGT. A user is authenticated this way by the Kerberos Server.
3. The Client uses the TGT to apply for application service tickets so that users can access specific applications. Each service ticket proves the Client's identity to an application server.
4. The Client presents the service ticket to the application server for authentication. The application server decrypts part of this ticket to check its authenticity.
5. If the application server finds that the service ticket is authentic, it applies the access control it previously defined for that client. If the application server cannot decrypt the service ticket, or if the service ticket has expired or is not authentic, the client is not authenticated.

The following sections describe a Kerberos session in more detail.

## Getting a Ticket-Granting Ticket From the Kerberos Server

The Kerberos Server has a secure database on its machine. A Client must get a ticket-granting ticket (TGT), which cannot be read by the Client, from the Kerberos Server.

The TGT lets a Client submit specific requests to the Kerberos Server for application service tickets that grant access to application servers. A Client must have an application service ticket when it requests a service from an application server.

The following process, as shown on figure 3.27, describes getting a TGT:

1. The Client user sends a request to the Kerberos Server. The request packet contains the client's user name.
2. The Kerberos Server looks for the user name in its secure database and extracts the private key for it.
3. The Kerberos Server:
  - a. Creates a randomly generated key to be used between the Client and the Kerberos Server. This is called the ticket-granting ticket's session key.
  - b. Creates a TGT that lets the Client obtain application service tickets from the Kerberos Server. The Kerberos Server encrypts this TGT using the private key obtained from the Kerberos database.

**Ticket:** {user-name, Kerberos Server name, Client Internet address, session key}private key

Kerberos also includes a timestamp in the TGT.

**Forms a packet containing the session key and the encrypted TGT, and encrypts the message from the Client's private key obtained from the secure database.**

**Packet:** {session key, encrypted ticket-granting ticket} Client private key

***Sends the packet containing the user's encrypted Kerberos password to the Client.***

4. The Client uses its private key to decrypt the packet. When the Client receives packet, the procedure prompts the Client for its password. Using the private key, Client encrypts the user's password and compares it to the encrypted password sent in the TGT. If the passwords match, the user has obtained a valid TGT; if not, the packet is discarded and the user cannot use Kerberos authentication to access any application servers.

## ***Getting Application Service Tickets for Network Services from the Kerberos Server***

Once a Client has a ticket-granting ticket, it can ask application servers for access to network applications.

Every request of this kind requires first obtaining an application service ticket for the particular application server from the Ticket-Granting Service (TGS).

Figure 3.28 and 3.29 outlined in the following process describe getting an application service ticket to use to access an application server.

### **The Client:**

1. Creates an authenticator to be used between the Client and the Kerberos Server. The Client encrypts the authenticator using the session key that it received previously. The authenticator contains three parts:
  - user name,
  - client Internet address,
  - current time
2. Creates the message to send to the Kerberos Server. The packet contains three parts:
  - ticket-granting ticket,
  - encrypted authenticator,
  - application server name
3. Sends the packet to the Kerberos Server. The Kerberos Server receives the packet from the Client.

### **The Kerberos Server:**

4. Decrypts the ticket-granting ticket using its private key to obtain the session key. (The ticket-granting ticket was originally encrypted with this same key.)
  - Decrypts the authenticator using the session key, which compares the:
    - User name in the ticket and authenticator
    - Kerberos Server name in the ticket and its own name
    - Internet address in the ticket, authenticator, and received packet
    - Current time in the authenticator with its own current time to make sure the message is authentic and recent.



After the Kerberos Server verifies the information in the ticket, the Server creates an application service ticket packet for the Client. The Server:

7. Uses the application server name in the message and obtains the application server's private key from the Kerberos database.
8. Creates a new session key and then an application service ticket based on the application server name and the new session key. The Kerberos Server encrypts this ticket with the application server's private key. This ticket is called the application ticket. This ticket has the same fields as the ticket-granting ticket:
  - user-name,
  - Application server name,
  - client Internet address,
  - new session key

Application server private key

2. Forms a packet containing the new session key and the encrypted application service ticket; encrypts the message with the session key, which the Client already knows:
  - new session key,
  - Application ticket

3. Sends the packet to the Client.

The Client decrypts the packet using the session key it received previously. From this message it receives the application service ticket that it cannot decrypt and the new session key to use to communicate with the application server.

Once a Client receives a ticket for an application service, the Client can request that service. The Client includes the application service ticket with the request for authentication that it sends to the application server. Figure 3.30 shows the process for requesting a service from an application server.

## ***Summary Of Kerberos Authentication***

There are three main steps in the Kerberos process. The Client:

1. Requests a ticket-granting ticket (TGT).
2. Presents the TGT and an authenticator to the Kerberos Server when it request access to an application server. The Kerberos Server grants the Client an application service ticket to access the application server.
3. Presents the application ticket and an authenticator to the application server when it requests access to the server. The server's access control policy either grants or denies access to services.

The Kerberos process uses tickets, authenticators, and messages. These elements provide specific encrypted information about clients and servers. Keys are used to encrypt and decrypt tickets, authenticators, and messages.

Some things to remember about tickets and authenticators:

- A Client must have a ticket-granting ticket and a service ticket to access any application server. The Client gets all tickets from the Kerberos Server.
- The Client cannot read tickets because the Kerberos Server encrypts them with the private key of the application server. Every ticket is associated with a session key.
- Every ticket-granting ticket has a lifetime (usually eight hours) and is reusable during that lifetime.
- Kerberos requires a new authenticator from the Client each time the Client starts a new connection with an application. Authenticators have a short lifetime (generally five minutes).
- The encrypted ticket and authenticator contain the Client's network address. Another user cannot use stolen copies without first changing his system to impersonate the Client's network address.

To hack Kerberos is very hard! In case of an attack, before the authenticator expires, a hacker would need to:

- steal the original ticket,

- Steal the authenticator,
- prevent the original copies of the ticket and authenticator from arriving at the destination server, and
- modify its network address to match the client's address.

## Cygnus' KerbNet

KerbNet security software is Cygnus' commercial implementation of MIT's Kerberos v5.

This is a great product to use when securing your network, as it provides the security of Kerberos, with its single trusted authentication server architecture, which provides the basis for a single sign-on interface for your users. Also, once you install and configure the KerbNet Authentication Server, client and server applications can be 'Kerberized' to work with KerbNet, which is very simple in a multi-user application environment. Basically, all you do is to replace your E-mail, ftp, or telnet, with Cygnus' off-the-shelf Kerberized versions.

The KerbNet libraries allows in-house developers to add KerbNet authentication and encryption directly to their existing client-server applications. KerbNet Authentication Server is the first Kerberos server for both UNIX and Windows NT with encrypted tickets for requesting services, which keeps passwords off the network, prevents password spoofing attacks, and allows for encrypted communications between a client and server.

### Tip:

For more information on KerbNet or to download a free copy, check Cygnus web site at URL <http://www.cygnus.com/product/kerbnet-index.html>

## Key-Exchange Algorithms (KEA)

Diffie-Hellman was the first one to invent a public-key algorithm in 1976. Instead of the calculating the exponentiation of a field, this public-key security scheme is very secure because it calculates the discrete logarithms in a finite field, which is very hard to do. Thus, Diffie-Hellman is ideal, and can be used, for key distribution. You can use this algorithm to generate your secret key. But don't get confused! You can't use it for encrypting or decrypting a message!

Lets take a look at how it works.

### Diffie-Hellman Public-Key Algorithm

Diffie-Hellman's system requires the dynamic exchange of keys for every sender-receiver pair. This two-way key negotiation is very good for enhancing the security of your messages. After you encrypt a message you can then use this scheme to further complicate the decryption of your message, as the hacker would have to decrypt the key, then the message. However, as you might imagine, this will require additional communications overhead.

In the RSA system, for example, communications overhead are reduced, as the ability to have static, unchanging keys for each receiver that are 'announced' by a formal 'trusted authority,' in this case the hierarchical model, or distributed in an informal network of trust.

Diffie-Hellman's method for key agreement actually has a simple math, which is aimed to allow two hosts to create and share a secret key. Assuming that you want to generate a secret key with your significant other (SO), here it is how this process works:

1. First, you and your partner must follow the "Diffie-Hellman parameters," which requires you to find a prime number "p," which should be larger then 2 and "base", "g," which should be an integer that is smaller than your prime number ("p"). You can either hard code them or fetch them from your server.
2. Both of you will have to each and secretly generate a private number that we will call "x," which should be less than (p-1).

- At this point, both of you will generate the public keys, which we will call "y." You guys will create them using the function:

$$y = g^x \% p$$

- You now will exchange the public keys ('y') and the exchanged numbers are converted into a secret key, "z."

$$z = y^x \% p$$

"z" now can be used as the key for whatever encryption method used to transfer information between the two of you. Mathematically speaking, you two should have generated the same value for "z," whereas

$$z = (g^x \% p)^{x'} \% p = (g^{x'} \% p)^x \% p$$

All of these numbers are positive integers, whereas

$x^y$  means: x is raised to the y power

$x\%y$  means: x is divided by y and the remainder is returned

#### Note:

The Diffie-Hellman Key Agreement, U.S. patent 4,200,770, is owned by Public Key Partners and will expire later this year, 1997.

## Cryptanalysis and Attacks

Cryptanalysis is the art of deciphering encrypted communications without knowing the proper keys, or if you prefer, it is the art of breaking the code! There are many cryptanalytic techniques, as well as cryptanalysts. Although it is true to say that every hacker trying to crack a code is a cryptanalyst, to say that cryptanalysis is threatening to security is not true. It is through cryptanalysis that one may find a weakness in the cryptosystem that could have eventually endangered the secrecy of a message being exchanged. So, crypto, in its many forms and shapes is indeed a double-edged sword.

An encrypted key can be compromised, if the key is exposed through a non-cryptanalytic way, such as if you were to write your public-key somewhere, so you will not forget it! If anyone finds it out, key would have just been compromised! But a key can be also attacked, if someone tries to apply cryptanalysis to it.

The next sections discuss some of the most typical cryptanalysis attacks.

### *Ciphertext-only Attack*

In this type of attack, the hacker, or cryptanalyst, does not know anything about the contents of the message, and must work from ciphertext only.

In practice it is quite often possible to make guesses about the plaintext, as many types of messages have fixed format headers. Even ordinary letters and documents begin in a very predictable way. It may also be possible to guess that some ciphertext block contains a common word.

The goal of the cryptanalyst here is then to try to deduce the key used to encrypt the message, which would also allow him/her to decrypt other messages encrypted with the same key.

## ***Known-plaintext Attack***

In this case, the hacker knows or can guess the plaintext for some parts of the ciphertext. The task is to decrypt the rest of the ciphertext blocks using this information. One way he will probably try is to determine the key used to encrypt the data.

## ***Chosen-plaintext Attack***

The hacker here is able to have any text he likes encrypted with the unknown key, he is able choose the plaintext that gets encrypted, which can lead him to ones that might yield more information about the key. Therefore, his task is to determine the key used for encryption. Some encryption methods, particularly RSA, are extremely vulnerable to chosen-plaintext attacks.

## ***Adaptive-chosen-plaintext Attack***

This is actually a variation of the chosen-plaintext attack. But in this case the hacker is able to exercise the option of modifying his choice of the encrypted plaintext based on the results of previous encryption, which allows him to choose a smaller text block of plaintext to be encrypted.

## ***Man-in-the-middle Attack***

This is a relevant attack for cryptographic communication and key exchange protocols. It's a sort of key spoofing, where a hacker would intercept the communication between two parties exchanging keys for secure communication, such as Diffie-Hellman's, corrupting the key by performing a separate key exchange with each party and forcing each one of them to use different keys, each of which is known by the hacker. The hacker will then decrypt any communications with a now valid key, and encrypt them with the other key for sending to the other party. Worse, the parties will still think that they are communicating securely, as this whole process is totally transparent to both parties, they would never know what has happened until it is too late!

One way to prevent man-in-the-middle attacks is that both sides compute a cryptographic hash function of the key exchange, use a digital signature algorithm, and send the signature to the other side. The recipient then verifies the authentication of the signature as being from the desired other party, and that the hash in the signature matches that computed locally.

## ***Chosen-ciphertext Attack***

In this case, the hacker, or cryptanalyst, not only is able to choose which ciphertext he will try to decrypt but also he/she has access to the decrypted plaintext.

Usually this type of attack is applied to public-key algorithms, which very often works well against symmetric algorithms too.

## ***Chosen-key Attack***

Although the name suggests that the attacker is able to choose the key, this is not true. As a matter of fact, this is a very weird form of attack where the hacker only has some knowledge about the relationship between the two keys. Bruce Schneier brilliantly discusses this form of attack in his book Applied Cryptography, in the section "Differential and Linear Cryptanalysis."

# Rubber-hose Cryptanalysis

This is the "dirty" way, where the hacker will harass, threaten, bribe and torture someone until they get the key!

## Tip:

For additional information on cryptanalysis attacks, check this references:

Bruce Schneier: Applied Cryptography. John Wiley & Sons, 1994.

Jennifer Seberry and Josed Pieprzyk: Cryptography: An Introduction to Computer Security, Prentice-Hall, 1989.

Man Young Rhee: Cryptography and Secure Data Communications. McGraw-Hill, 1994.

M. E. Hellman and R. C. Merkle: Public Key Cryptographic Apparatus and Method.

The RSA Frequently (<http://www.rsa.com/faq.htm>) Asked Questions document by RSA Data Security, Inc., 1995.

## Timing Attack

This is somewhat a new form of attack discovered by Paul Kocher that looks at the fact that different modular exponentiation operations in RSA takes discretely different amounts of time to process. In this process, the cryptanalyst repeatedly measures the exact execution times of modular exponentiation operations, which is very relevant for RSA, Diffie-Hellman, and Elliptic Curve methods.

Usually, RSA computations are done with what is called Chinese Remainder theorem (see figure 3.31). But if it doesn't, a hacker could exploit slight timing differences in RSA computations in order to try to recover it.

Figure 3.31 shows a description of the Chinese Remainder Theorem at SouthWest Texas State University Web site.

## Tip:

To learn more about the Chinese Remainder Theorem, check the URL [http://www.math.swt.edu/~haz/prob\\_sets/notes/node25.html](http://www.math.swt.edu/~haz/prob_sets/notes/node25.html), at SouthWest Texas State University.

The attacker passively observes "k" operations measuring the time "t" it takes to compute each modular exponentiation operation:  $m=c^d \bmod n$ . The attacker also knows "c" and "n." The pseudo code of the attack is:

Algorithm to compute  $m=c^d \bmod n$ :

Let  $m_0 = 1$ .

Let  $c_0 = x$ .

For  $i=0$  upto (bits in  $d-1$ ):

If (bit  $i$  of  $d$ ) is 1 then

Let  $m_{i+1} = (m_i * c_i) \bmod n$ .

Else

Let  $m_{i+1} = m_i$ .

Let  $d_{i+1} = d_i^2 \bmod n$ .

End.

According to Ron Rivest ([rivest@theory.lcs.mit.edu](mailto:rivest@theory.lcs.mit.edu)), at MIT, the simplest way to defeat this timing attack would be to "ensure that the cryptographic computations take an amount of time that does not depend on the data being operated on. For example, for RSA it suffices to ensure that a modular multiplication always takes the same amount of time, independent of the operands."

He also suggest a second alternative, using "blinding techniques. According to him, you could "blind the data beforehand, perform the cryptographic computation, and then unblind it afterwards. For RSA, this is quite simple to do. (The blinding and unblinding operations still need to take a fixed amount of time.) This doesn't give a fixed overall computation time, but the computation time is then a random variable that is independent of the operands."

**Note:**

This blinding process introduces a random value into the decryption process, whereas,

$$m = c^d \bmod n$$

becomes:

$$m = r^{-1}(cr^e)^d \bmod n$$

$r$  is the random value, and  $r^{-1}$  is it's inverse.

The University of British Columbia (<http://www.ubc.ca/>) has a Web site at URL <http://axion.physics.ubc.ca/pgp-attack.html> with vast documentation of symmetric and asymmetric crypto attacks, which are well worth checking out.

# Cryptography Applications and Application Programming Interfaces (APIs)

If you really want to understand what is going on in the crypto world you will need to grasp the ever-increasing progress and development of new applications applied to the flow of information on electronic highways, the need for secure and private communication and... control.

Of course, the government shares the concern when electronic transfer of money and the transmission of commercial information is taking place on the Internet more and more often. Although the government does care, its controversial proposal to address the security of electronic transactions over the Internet, known as the "Clipper chip" proposal, is also a double-edged sword, as it does offer secure transactions, but in a government controlled tappable way.

The result of this frenzy is the increased development of many cryptographic applications and application programming interfaces (APIs). Data privacy and secure communication channels, which includes but are not limited to authentication mechanisms and secure standards begins to be developed and proposed.

The George Washington University (<http://www.seas.gwu.edu/>), through the Cyberspace Policy Institute (<http://www.cpi.seas.gwu.edu/>) has a great selection of information policy bibliography that offers a solid foundation for the need for data protection, secure communications and its implications in the whole information processing. This section discusses some of these efforts and their impact in the security of the Internet, the hole of cryptography and firewalls.

## *Data Privacy and Secure communications channel*

Internet users, as well as protected network users should always be responsible for data privacy within the organization and data exchanges. But it is the responsibility of the Internet manager to make sure an Internet security policy, outlining the privacy of information exists so that users can be held accountable for following it.

It should be also his/her responsibility to protect the personal privacy of users, as clearly stated in the policy, which should identify the elements of the company's structure, such as the different levels of data confidentiality (some data may need to be encrypted, while other may need some sort of access control) and practice.

Data security policy should be applied throughout the company, regardless of the nature of the data, the storage form or location. Users must understand that the protection of individual privacy and information will only occur if all users are committed, by knowing and respecting the security policy in place. Thus, recipients of confidential data and files downloaded directly to their computers should preserve the confidentiality of data.

Also, application developing groups must take in consideration the security policy in place, and at least, if no policy is in vogue, that the applications developed do take in consideration security aspects, both at the Intranet and the Internet level.

More about security and firewall policies will be discussed later on chapter 9, "Setting up a Firewall Security Policy." For now, let's look into the authentication processes and security API and how they impact the secure communication channels.

## **Some Data Privacy Prime and Tools**

There are several applications already available on the market to work as a standalone or in association with other applications or devices, such as firewalls, proxies and routers, to protect your personal and financial private data. Pretty Good Privacy (PGP) and Privacy Enhancement Mail (PEM) are few examples, but first things first!

You first need to understand your system. A logical or physical map can be essential to understanding the vulnerable points to data security of your system. A map also will help you in planning future changes to a system. Planning these future changes will ensure a tight grip on the reigns of security and will increase your awareness of the questions that need answering. Such control becomes critical when implementing a new addition to a system, such as a new network service like anonymous FTP (if the site does not offer it already).

## ***Have a Password Policy***

The easiest and, believe me or not, most secure way to protect an user is through a good password policy, which will force everyone on a Intranet/Internet environment to be authenticated. Nevertheless, if good passwords are vital for data security, passwords are also the first target of attack! This because many users and network systems make it easier to attack a password than attack other possible security holes in the system.

Anyone has the potential to obtain information that leads to the discovery of your password, as there are many tools, such as Crack, NTCrack and many others developed to help on cracking passwords. But many times, password information is giving away too casually. It easily can be one friend giving another the use of their account or someone watching over a shoulder while a password is entered. The best way to combat this is to choose a unique password containing a combination of letters and numbers, preferably not found in a dictionary, for example, and to be very conscious of who can observe it.

Therefore, the following is a list of tips on creating unique and safe password:

- Use numbers in the password, preferably not at the beginning or ending.
- Use non-basic words in the password. Basic words in foreign languages are just as easy to guess.
- Try keyboard tricks like shifting fingers to the left or right one key when typing the password.
- If someone is standing over your shoulder, you could always politely ask him to turn his head.
- Use non-alphanumeric characters in the password. Symbols such as \$, %, ^, and & are often valid characters to use in passwords.
- Administrators can use control characters, on certain systems, in the middle of the password. You can determine which control characters can be used by trial and error.
- Use a mixture of uppercase and lowercase letters.

## ***Authentication***

As discussed above, a good password policy is very important to safeguard the integrity, confidentiality and security of your users, specially if you are involved with electronic commerce, which becomes a requirement. Therefore, authentication must become a daily user's process, rather than a special procedure, for users to logon at their computers, the Internet and departmental Intranets.

When applying authentication methods, it is important to take into consideration the spoofing risks. Cryptography methods, as discussed earlier will help you to implement a security policy not so easy to be spoofed by hacker, but it may not be enough to protect corporate resources and other non-individual related data or resource.

Therefore, you must incorporate other strategies and technologies to enhance the level of security of your corporate network. Firewalls are definitely a requirement and will be discussed in more details throughout this book.

## ***Authenticode***

Authenticode appeared as one of Microsoft's earliest commercial implementations of code signing. The Authenticode signature is based on currently prevailing industry standards - X509v3 certificates and PKCS#7 signature blocks.

### **Tip:**

Documentation on Authenticode and related infrastructure can be found at <http://www.microsoft.com/intdev/security>.

That has being a lot of commentary in the USENET and trade magazines about ActiveX and Authenticode, but most of them focusing on how an ActiveX control operates and what Microsoft should or should not have included with the tool. So I don't intend to reproduce the same line of thoughts here as if you want to know more about the hows, dos and don'ts of



Authenticode probably a search on AltaVista would be enough. Rather, I would like us to focus on the infrastructure of what Microsoft proposes with Authenticode and its impact as a data security cryptographic-based application.

Brent Laminack (brent@ecweb.com) posted the following considerations about Authenticode on the USENET, which clearly illustrates a basic infra-structure issue with it. You judge for yourself if you would be willing to base your data security tasks to it.

Laminack suggests us to consider two ActiveX controls. One providing a control similar to the Win95 "Start" button with all the commands on the user's computer presented in a list to choose from. Suppose it keeps these command names in a preferences file such as C:\windows\mycommands. The file may contain a list such as: Word, Excel, format c:, IE3, etc.

He also suggests us to consider a second ActiveX control that provides a "cron" facility, which would automatically wake up at a specified time and execute a list of commands for housekeeping such as backup, defrag, etc. Suppose it keeps its list of commands in, say, for instance C:\windows\mycommands. In his own words, "you see it coming," don't you? What could happen is that the second control could find the file written by the first one and dutifully fire up Word, Excel, and then... format the C drive. Commands after this one are of diminishing consequence.

What now? You're stuck! You now have a wiped hard drive and, as Laminack puts it, you have no fingerprints for Authenticode. Even if you do get them, who are you going to sic the law enforcement people on? Both controls did exactly what they were designed to do, exactly what they advertised to do. Who are you going to sue?

Worse, neither of the codes "misbehaved." What did in your disk was an unforeseen interaction between the two. Laminack suggests that with a bit of thought work it would be possible to come up with a co-operating gang of ActiveX controls to do deliberate theft via collusion where each program is only doing what it's "supposed" to, yet the total of their activity is much greater than the sum of the parts. Yes, non-linearity is clearly at work here in the interaction of the components.

The only way to avoid this would be to strictly decouple them, by not allowing any to share information with the other, such as giving each its own private file-space to write in. This, alas is not the case.

As Microsoft puts it, the way Authenticode is implemented, both contractually and technically, at least in its present release (March of 1997), when you sign a code you are actually taking explicit responsibility as the code's publisher, an action not to be taken lightly from a legal point of view.

But it is just too easy to say that by signing a code gives you accountability. After all, would you have an audit trail to use as supporting evidence? Also, in the software industry, history shows that usually a piece of software is not liable for damages it may cause to a system!

Although Authenticode is still the only deployed code-signing application, Netscape Navigator 4.0 already has code signing, and JavaSoft's JDK1.1 as well. The bottom line? You'll need much more than Authenticode, and Microsoft's response is SSPI

## ***NT Security Support Provider Interface (SSPI)***

Microsoft's Security Support Provider Interface (SSPI) is a common Application Programming Interface (API) for obtaining integrated security services for authentication, message integrity, message privacy, and security quality of service for any distributed application protocol. Application protocol designers can take advantage of this interface to obtain different security services without modification to the protocol itself.

Figure 3.32 shows where the SSPI security services fit into the overall distributed application architecture.

SSPI provides a common interface between transport-level applications, such as Microsoft RPC or a file system redirector, and security providers, such as Windows NT Distributed Security. SSPI provides a mechanism by which a distributed application can call one of several security providers to obtain an authenticated connection without knowledge of the details of the security protocol.

SSPI consists of following APIs:

- **Credential Management APIs** — which provides access to credentials (password data, tickets, and so on) of a

principal or free such access. The APIs are:

- **AcquireCredentialsHandle** — Acquires a handle to the reference credentials.
- **FreeCredentialsHandle** — Releases a credential handle and associated resources.
- **QueryCredentialAttributes** — Allows queries on various credential attributes like associated name, domain name, and so forth.
- **Context Management APIs** — Context management APIs provide methods for creating and using security contexts. The contexts are created on both the client and the server side of a communication link. These contexts can then be used later with the message support APIs. The APIs are:
  - **InitializeSecurityContext** — Initiates a security context by generating an security token that can be passed to the server.
  - **AcceptSecurityContext** — Creates a security context using the opaque message received from the client.
  - **DeleteSecurityContext** — Frees a security context and associated resources.
  - **QueryContextAttributes** — Allows queries on various context attributes.
  - **ApplyControlToken** — Applies a supplemental security message to an existing security context.
  - **CompleteAuthToken** — Completes an authentication token, since some protocols, like DCE RPC, need to revise the security information once the transport has updated some message fields.
  - **ImpersonateSecurityContext** — Attaches the client's security context as an impersonation token to the calling thread.
  - **RevertSecurityContext** — Ceases impersonation and defaults the calling thread to its primary token.
- **Message Support APIs** — Message support APIs provide communication integrity and privacy services based on a security context. The APIs are:
  - **MakeSignature** — Generates a secure signature based on a message and a security context.
  - **VerifySignature** — Verifies that the signature matches a received message.
- **Package Management APIs** — Package Management APIs provide services for different security packages that the security provider supports. The APIs are:
  - **EnumerateSecurityPackages** — Lists available security packages and their capabilities.
  - **QuerySecurityPackageInfo** — Queries an individual security package for its capabilities.

SSPI does not currently provide any public interfaces for encryption/decryption functionality. A *security provider* is a dynamic-link library that implements the Security Support Provider Interface and makes one or more *security packages* available to applications. A security package maps the SSPI functions to an implementation of the security protocol specific to that package, such as NTLM, Kerberos, or SSL. Security packages are sometimes referred to as "SSPs," such as the "NTLM SSP." The name of the security package is used in the initialization step to identify a specific package.

The Security Support Provider Interface allows an application to use any of the available security packages on a system without changing the interface to use security services. SSPI does not establish logon credentials because that is generally a privileged operation handled by the operating system.

An application can use the package management functions to list the security packages available and select one to support its needs. The application then uses the credential management functions to obtain a handle to the credentials of the user on whose behalf they are executing. With this handle, the application can use the context management functions to create a security context to a service. A *security context* is an opaque data structure that contains the security data relevant to a connection, such as a session key, the duration of the session, and so on. Finally, the application uses the security context with the message support functions to ensure message integrity and privacy during the connection.

## Microsoft Cryptographic API (CryptoAPI)

The Microsoft Cryptographic API (CryptoAPI) provides services that enable application developers to add cryptography to their Win32 applications. Applications can use the functions in CryptoAPI without knowing anything about the underlying implementation, in much the same way that an application can use a graphics library without knowing anything about the

particular graphics hardware configuration.

CryptoAPI is a set of functions that allow applications to encrypt or digitally sign data in a flexible manner, while providing protection for the user's sensitive private key data.

All cryptographic operations are performed by independent modules known as cryptographic service providers (CSPs). One CSP, the Microsoft RSA Base Provider, is bundled with the operating system.

Each CSP provides a different implementation of the CryptoAPI. Some provide stronger cryptographic algorithms while others contain hardware components such as smartcards. In addition, some CSPs may occasionally communicate with users directly, such as when digital signatures are performed using the user's signature private key.

**Note:**

For more and detailed information about CryptoAPI, check the URL  
<http://www.graphcomp.com/info/specs/ms/capi.html>.

## ***Cryptography and Firewalling: The Dynamic Dual***

No doubt, companies want, and need, a piece of the Internet. For some of you, Internet Managers, this may involve implementing an Intranet--a private IP network created with Web servers and browsers that runs over your protected network. But most likely, it will involve setting up the meaning for transferring data, including sensitive one, over the Internet.

Firewalls play a major role in protecting corporate sites from the Internet, but the old firewall concept, based on routers and few deny/allow statements are no longer enough to keep the hackers and crackers out. The statistics are not encouraging, as according to the Computer Security Institute, 1/5 of all companies on the Internet have been or are going to be hacked. Worse, at least 1/3 of them will be hacked after a firewall has being in place!

Chapter 7, "What is an Internet/Intranet Firewall After All" provides in depth details of the various types of firewalls available on the market, its features and, mostly important, the technology behind it. However, data can be stolen on the Internet, despite the presence of a firewall, as they can be intercepted outside the firewall, while still on the Internet. Besides, as data can be stolen on the Internet, it can also be modified. Anyone could, for example, insert an malicious applet as an attachment to an intercepted e-mail message, that once activated it could disable the firewall or even compromise the security of your protected network.

Consider this scenario: you contract me to develop some applications using ActiveX. I develop some applications as plug-ins for you Internet Explorer and you get all happy. However, once your users agree to use this plug-in I become registered with Explorer as a trusted publisher. What it means is that from now on all the requisitions to download the plug-in I developed won't trigger the permission dialog box! Is it a bug or a feature?! Remember the ActiveX discussion earlier?

Far from being a fiction, unfortunately it is real. If you check C|net's URL at <http://www.news.com/News/Item/0,4,3707,00.html>, you see that earlier this year, around February of 1997, the same thing happened to InfoSpace. Fortunately, InfoSpace folks saw this "resource" as a bug and did an update on their plug-in. But here it is the question: can we assume that all the plug-in editors for the Internet Explorer are as responsible as InfoSpace?

When a download of an executable component is done, this component shouldn't be able to silently manipulate the security policy of a system, especially since the firewall, if any present, could not stop the corrupted message from accessing the protected network. However, it is almost impossible to prevent such a behavior from happening when we consider the active content model of Microsoft.

It is not new that the Java model is more robust then ActiveX when addressing this problem. But as a side effect, Java lacks such a feature, well, if we consider it a feature!

A feature or a bug, what I am most concerned is the fact that a shrewd developer could generate an ActiveX control that would do nothing more than open the doors of the system and let all the other programs come in without even passing through the Authenticode. This ActiveX control could even let another version of itself access the system, accordingly signed, but without malicious codes, which would cover up any trace of it in the system.

Unfortunately, with ActiveX, when an user allows the code to run on the system, there are many "distressing" situations that could happen. In a way, this is not a problem affecting only ActiveX. It extends through all the platforms and type of codes. If the Web made it easy for an editor to distribute his codes, it also made it easy to identify a malicious code and to alert and communicate the endangered parties.

Without a doubt, the Authenticode helps a lot in the quality control and authenticity of the code. The fact that we can rapidly identify the author of a code and demand from him a fix for a bug is an example of it. If the author refuses to fix the code, there are several avenues one could take to force him to fix it, both in the commercial level, refusing to use the code, as well as legally, bringing him to court. This feature alone already grants Authenticode some merit.

Even though, Java's robustness and the existence of other security applets for Java, such as Java Blocking, for instance, is enough for one to argue on rather develop on ActiveX or Java.

One alternative to prevent such a vulnerability is to run a filter in combination to the firewalls, so that these applets (Java, JavaScript or ActiveX objects) can be filtered. A major example of such a tool is the so called Java Blocking, which have created a lot of confusion as far as how to run it in the most effective way, as opinions are many.

My recommendation is to run the Java Blocking as a service at the firewall. This way, it will extend the level of protection against Java applets throughout the whole network. Some browsers, such as Netscape Navigator, provide security against Java applets at the client level, allowing the user to disable Java applets at the browser. However, it becomes very difficult to administer all the clients centrally.

Carl V Claunch, from Hitachi Data Systems, developed a patch for the TIS firewall toolkit that converts the TIS http-gw proxy onto a proxy filter. This filter can be implemented as an uniform or differentiated security policy at the level of IP/domain addresses. This filter can block, permit or combine both instances based on the browser version. The security policies are created separately for Java, JavaScript, VBScript, ActiveX, SSL e SHTTP

According to Claunch, as far as blocking JavaScript, this process involves the scanning of various constructs:

- 1 - `<SCRIPT language=javascript> ... </SCRIPT>`
- 2 - `<SCRIPT language=livescript> . . . </SCRIPT>`
- 3 - Attribute in other tags on form onXXXX= where XXXX indicates the browser's actions, such as click, mouse movements, etc.
- 4 - URLs at HREFs and SRCs with javascript: protocol
- 5 - URLs at HREFs and SRCs with a livescript: protocol

The Java Blocking consists in disactivating both tags `<APPLET ...>` and `</APPLET>`, while allowing characters to pass, which usually are alternatively HTML.

AS for the VBScript blocking it involves:

1 - The scanning and filtering sequence of <SCRIPT language=VBScript> ....

</SCRIPT>

2 - Scanning and filtering sequence <SCRIPT language=vbs>...</SCRIPT>

3 - Removal of attributes on form onXXXXX= and many tags, just like with JavaScript

The blocking of ActiveX involves the removal sequence of <OBJECT...>...</OBJECT>.

However, the dialogs of SSL and SHTTP turns HTML blurry to the proxy. Consequently, these shttp and https HTML pages can't be effective filtered!

But don't you think that I'm hammering ActiveX and promoting Java! Anyone could develop a malicious plug-in for Netscape if they wanted to. As a matter of fact, the impact would have been even greater then with any ActiveX object when we consider the browsers. After all, a plug-in has as much control over Windows as an ActiveX object.

Don't even tell me that the advantage is in having to install a plug-in versus automatically receiving an ActiveX object. There are so many implementations of Netscape out there that for sure there would had been so many users installing such a malicious plug-in as ActiveX users facing a malicious ActiveX on their pages. Furthermore, you have no way to better control the installation of a plug-in on Netscape better then you would control the installation of an ActiveX object.

As professionals involved with network and site security, lets be realistic. Many experts have been pointing out the security flaws existent on Java implementations, as well as fundamental problems with the Java security model. As an example, I could cite attacks that confuses Java's system, resulting in applets executing arbitrary codes with total permission from the user invoking the applet.

**Note:**

There is a white paper, written by Dean, Felten e Wallach, entitled "Java Security: From HotJava to Netscape and Beyond" that discusses most of the problems and security flaws of Java. The paper is available for download at Princeton University's site, at URL <http://www.cs.princeton.edu/sip>.

So far, users and systems developers have been content in considering these Java problems... "temporary." They have been confident that bugs will be fixed quickly, limiting the margin of damages. Netscape has been incredible quick in fixing serious problems!

However, with the huge base of browsers capable of running Java, each one inviting a hostile applet to determine the actions of this browser, gives as the suspicion of a security flaw on Java at the implementation structure level.

There is another paper, available at Boston University's URL at <http://www.cs.bu.edu/techreports/96-026-java-firewalls.ps.Z>, that describes attacks to firewalls that can be launched from legitimate Java applets. The document describes a situation where in some firewall environments, a Java applet running on a browser inside the firewall can force the firewall to accept connections such as TELNET, or any other TCP ones, directed to the host! In some cases, the applet can even use the firewall to arbitrarily access other hosts supposedly protected by a firewall.

Let me explain that the weaknesses exploited in these attacks are not caused by Java implementations themselves, nor by the firewall itself, but from the combination of both elements together, and on the security model resulted from the browsers

access to hosts supposedly protected.

For those skeptical about the security of Java applets running on the Web, especially on browsers, here it is a "St. Thomas" test: for those running Netscape 3.0, check the URL at <http://www.geocities.com/CapeCanaveral/4016/>. Once there, check the Java-Jive page and watch the "little Java devils" working!

If you didn't realize what happened, try again and pay attention: every time you enter the page, a message is sent to the author of that page, Francesco Iannuzzelli ([ianosh@mv.itline.it](mailto:ianosh@mv.itline.it)), this without even asking your permission! The message he receives will contain your address (both user and SMTP server!) as you specified on your Netscape "preferences." According to Iannuzzelli, there is no way for you to be alerted about this bug!

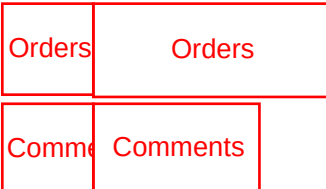
The only way you would have noticed something different was going on would had been the button you see on the page, which can be hidden, and the status bar showing a connection to your mail server, which can be hidden as well!

What to do then? Encryption is the obvious alternative. The great news is that firewall vendors are realizing that an offering encryption features with their firewall products. Many are even including applet filters. Vendors like Border Network Technologies Inc., Check Point Inc. and Trusted Information Systems Inc. are some of them.

Router vendors are also working hard on increasing the level of protection they can offer to corporate networks through their products. Cisco Systems Inc. and Network Systems Corp. are some examples.

According to Lee Bruno, in an article for Data Communications on the Web, back on April of 1996, mentioned few companies already offering standalone encryption devices. As Bruno suggests, "choosing the right gear means grappling with some complex issues. Start with the basics: Where is the data being encrypted? Some vendors do it at the application level; others, in the IP stack. The former lets net managers pick and choose what they want to encrypt. The latter forces them to encrypt everything on a given link."

The underlying truth here is that encryption and firewalling becomes a dynamic dual. You should review the information and recommendations on the whole Part II of this book, "Firewall Implementations and Limitations," consider what we discussed in this chapter about cryptography and its applications and build your own security policy. Keep in mind that you will need both, encryption and firewalls, to soundly protect your corporate network. Don't forget to read the Part III of this book, "Firewall Resource Guide," which will help you to run a "firewall attack drill" and provide you with abundant information on firewall vendors, utilities and complementary information.



---

[COMPUTING MCGRAW-HILL](#) | [Beta Books](#) | [Contact Us](#) | [Order Information](#) | [Online Catalog](#)

---



HTML conversions by [Mega Space](#).

This page updated on December 05, 1997 by [Webmaster](#).

[Computing McGraw-Hill](#) is an imprint of the [McGraw-Hill Professional Book Group](#).

Copyright ©1997 [The McGraw-Hill Companies, Inc.](#) All Rights Reserved.

Any use is subject to the rules stated in the [Terms of Use](#).

|        |        |
|--------|--------|
| Orders | Orders |
|--------|--------|

|      |         |
|------|---------|
| Back | Forward |
|------|---------|

|          |          |
|----------|----------|
| Comments | Comments |
|----------|----------|

© 1997 The McGraw-Hill Companies, Inc. All rights reserved.  
Any use of this Beta Book is subject to the rules stated in the [Terms of Use](#).

---

# Chapter 4

## *Firewalling Challenges: The Basic Web*

This chapter discusses the challenges firewall implementations face in light of the HyperText Transmission Protocol (HTTP) and some of its security issues. It also discusses the proxying characteristics of HTTP and its security concerns. It explores the secure HTTP (S-HTTP) as well as the use of SSL for enhanced security and reviews the security implications of Common Gateways Interface (CGI)

## HTTP

Being an application-level protocol developed for distributed, collaborative, hypermedia information systems, the Hypertext Transfer Protocol (HTTP) is a very generic and stateless protocol, enabling systems to be built independently of the data being transmitted. It is also an object-oriented protocol with capabilities to be used for a variety of tasks, which includes but is not limited to name servers, distributed object management systems and extension of its request methods, or commands.

One of the great features of HTTP is the typing and negotiation of data representation. This protocol has been in use since 1990, with the W3 global information initiative.

The most current version of HTTP is version 1.0, which is supported by all Web servers in the market. But there is also another version of the protocol, HTTP-NG (Next Generation), which promises to use the bandwidth available more efficiently and enhance the HTTP protocol.

Further, HTTP is a protocol that can be generically used for communication between user agents and proxies or gateways to other Internet protocols, such as SMTP, NNTP, FTP, Gopher and WAIS.

Nevertheless, all this flexibility offered by HTTP comes at a price: it makes Web server, and clients, very difficult to secure. The openness and stateless, characteristics of the Web, accounts for its quick success, but makes it very difficult to control and protect.

On the Internet, HTTP communication generally takes place over TCP/IP connections. It uses as default port 80, but other ports can be used, which does not prevent HTTP from being implemented on top of any other protocol. In fact, HTTP can use any reliable transport.

When a browser receives a data type it does not understand, it relies on additional applications to translate it to a form it can understand. These applications are usually called viewers, and should be the one of the first concerns you should have when preserving security. You must be careful when installing one, because, again, the underlying HTTP protocol running on your server will not stop the viewer from executing dangerous commands.

You should be especially careful with proxy and gateway applications. You must be cautious when forwarding requests that are received in a format different than the one HTTP understands. It must take into consideration the HTTP version

in use, as the protocol version indicates the protocol capability of the sender. A proxy or gateway should never send a message with a version indicator greater than its native version. Otherwise, if a higher version request is received, both the proxy or the gateway must either downgrade the request version, respond with an error, or switch to a tunnel behavior.

**Note:**

If you need more information on HTTP, check the URL:  
<http://www.w3.org/hypertext/WWW/Protocols/>

There is a series of utilities intended for Web server administrators available at the URL:  
<ftp://src.brunel.ac.uk/WWW/managers/>

The majority of HTTP clients, such as Purveyor (<http://www.process.com>) and Netscape Navigator, support a variety of proxying schemes, SOCKS and transparent proxying.

Purveyor, for instance, provides proxy support for not only HTTP, but also FTP and GOPHER protocols, creating a secure LAN environment by restricting internet activities of LAN users. The proxy server offers improved performance by allowing internal proxy caching. Purveyor also provides proxy-to-proxy support for corporations with multiple proxy servers.

**Tip:**

For more information on Purveyor Webserver, check Process Software's URL:  
<http://www.process.com>.

If you are running your Web server on Windows NT, Windows 95 or NetWare, you can use Purveyor Webserver's proxy features to enhance security. In addition, you can increase the performance of your server as Purveyor can locally cache Web pages obtained from the Internet.

Installing a firewall at your site should be a must. Regardless if you are placing your server outside or inside your protected network, a firewall will be able to stop most of the attacks, but not all! The openness of HTTP is too great for you to risk. Besides, you still have all the viewers and applets to worry about.

When selecting a firewall, make sure to choose one that includes HTTP proxy server, check Appendix A "Types of Firewalls and Products on the Market" for a complete review of all the major firewalls vendors and specifications of their products. Also, check the CD that accompanies this book, as many of the vendors listed on Appendix A provided demos and evaluation copies of their products, which are worth testing.

Firewalls will tremendously help you protecting your browsers. Some firewalls, such as TIS FWTK provide HTTP proxying totally transparent to the user. More will be seeing about firewall in chapter 7, "What is an Internet/Intranet Firewall After All." For now, you must be aware of the firewalling challenges when dealing with Web security requirements and the HTTP protocol.

## ***The Basic Web***

Do you know what happens when a user connects to your site? If you don't know how they come in, you will not know how to lock the door.

If you have a Web server on your site, every time a user establishes a connection to it, his client passes to your Web server the numeric IP address of the machine. In some wise situations, the IP address your Web server will receive is



not even the client's address, but the address of the proxy server his requests goes through. What your server will see then is the address of the proxy requesting the document on behalf of the client. But the client, thanks to the HTTP protocol, can also disclose to the Web server the username logged at the client, making the request.

Unless you have set your server to capture such information, what it will do first is to reverse the numeric IP address in an attempt to get the domain name of the client (e.g. www.vibes.com). But in order for the server to get this domain name, it must first contact a domain name server and present it with the IP address to be converted.

Many times, IP addresses cannot get reversed as they were not correctly configured. Consequently, the server cannot reverse the address. What happens next? The server goes ahead and forges the address!

Once the Web server has the IP address and the possible domain name for the client, it starts to apply a set of authentication rules, trying to determine if the client has access permission to the document requested.

Did you notice the security hole? There are few security holes here, as a result of this transaction:

- The client requesting the information may never get it as the server had forged the domain name. The client now may not be authorized to retrieve the information requested,
- The server may send the information to a different client as the domain name was forged.
- Worse, the server may allow access to an intruder under the impression it is a legitimate user!
- The risks here go both ways:
- You should be concerned with the HTTP server and what risks, or harm it can bring to your clients, but also
- You should be concerned with the HTTP clients and what risks, or harm it can bring to your server.

As discussed above, as far as client's threats to your server, you should be careful with the security of your server. You should make sure clients will access only what they are supposed to and if there is a hostile attack, that your server has some way to protect the access to it.

However, not all is loss, as there are few basic steps you can follow in order to enhance the security of your server:

- Make sure to configure your server carefully, and to use its access and security features.
- You can also run your Web server as an unprivileged user.
- If you are running your server on a Windows NT system, make sure to check the permissions for the drives and shares and to set the system and restricted areas read-only. Or you can use chroot to restrict access to the systems section.
- You can mirror your server and put sensitive files on the primary system but have a secondary system, without any sensitive data open for the Internet.
- Remember Murphy's Law, whatever can go wrong, WILL go wrong. Expect the worse and configure your Web server in a way that even if a hacker is to take total control over it, there is going to be a huge wall (if not a firewall!) to be crossed.
- Most important, review the applets and script your HTTP server uses, especially those CGI scripts interacting with your clients over the Internet. Watch for possibilities of external users triggering execution of inside commands.
- Run your Web server on a Windows NT server. It is much more secure, although it may not have as much features as the UNIX and Suns counterparts.
- Macintosh Web server are even more secure, but lack on implementation features when compared with Windows NT and 95 platforms.

To illustrate what a misconfigured domain name can do to a reversal IP address process, take in consideration the entries you enter in your access.conf file. Keep in mind that this file is responsible for the access control of the documents in your server.

When setting up this file, you will need to put a <directory> tag, for each directory being controlled, into the

access.conf file. Within the <directory> tag you will also need to use a <limit> tag with the parameters (allow, deny, and order) needed to control access to the directory.

The following is an example where the whole Cyberspace can access the files in you top-level document directory:

```
<directory /usr/local/http/docs>
```

```
<limit>
```

```
order allow,deny
```

```
allow from all
```

```
</limit>
```

```
</directory>
```

One of the key lines here is the "order" directive, telling the server to process "allow" directives (from ALL clients) before any "deny" directives. Have you noticed we don't have any "deny" directive?

Now lets assume you need to restrict an area on your server only for internal users to access it. Unlike the above example, you will need a "deny" directive:

```
<directory /usr/local/http/docscorp>
```

```
<limit>
```

```
order deny,allow
```

```
deny from all
```

```
allow from .greatplace.com
```

```
</limit>
```

```
</directory>
```

In this case, the "deny" directive came before the "allow" directive, so that the whole Cyberspace can have its access restricted access to the company. The "allow" directive permit access from anyone coming from greatplace.com domain.

If the server can't reverse the IP address of a client, then you have a problem, as the domain name is critical to this process. Simply put, the user will not be able to access the Web page.

But, there it is a "Band-Aid" solution. You can add raw IP numbers to the access list.

```
<directory /usr/local/http/docscorp>
```

```
<limit>
```

```
order deny,allow
```

```
deny from all
```

```
allow from .greatplace.com 198.155.25
```

```
</limit>
```

```
</directory>
```

This way, the directive "allow" will permit any access coming from "greatplace" but also from any machine where the IP address starts with 198.155.25.

## ***What to Watch for on the HTTP Protocol***

The HTTP protocol has some more security holes to justify a firewall. One of them is that it allows remote users to request communication to a remote server machine, and to execute commands remotely. This security hole compromises the Web server and the client in many ways, including but not being limited to:

- Arbitrary authentication of remote requests.
- Arbitrary authentication of Web servers.
- Breach of privacy of request and responses.
- Abuse of server features and resources.
- Abuse of servers by exploiting its bugs and security holes.
- Abuse of log information (extraction of IP addresses, domain names, file names, etc.)

Most of these security holes are well known. Some applications like Netscape's SSL and NCSA's S-HTTP try to address the issue, but only partially.

The problem is that Web servers are very vulnerable to client's behavior over the Internet. I recommend you to force Web clients to prompt a user before allowing HTTP access to reserved ports other than the port reserved for it. Otherwise, these could cause the user to unadvertedly cause a transaction to occur in a different and danger protocol.

Watch the GET and HEAD methods! The so trivial link to click an anchor to subscribe or reply to a service can trigger an applet to run without the user's knowledge, which enables the abuse by malicious users.

Another security hole of HTTP has to do with server logs. Usually, a Web server logs a large amount of personal data about information requested by different users. Evidently, this information should remain confidential. HTTP allows the information to be retrieved without any access permission scheme.

There is a feature, the "Referer:" field, that increases the amount of personal data transferred. This field allows reading patters to be analyzed and even reverse links to be drawn. If in wrong hands, it could become a very useful and powerful tool that can lead to abuse and breach of confidentiality. To this day, there are cases where the suppression of

the Referer information is not know. Developers are still working on a solution.

Many other HTTP limitations and security holes exist if we were to break down the ramifications of the above security issues presented by the protocol. Secure HTTP technologies and schemes are an attempt to address and resolve these security holes.

## ***Taking Advantage of S-HTTP***

Secure HyperText Transfer Protocol (S-HTTP) was developed to fill the gap of security in protecting sensitive information as it is transmitted over the Internet. As the need for authentication among Internet and Web grows, users need to be authenticated before sending encrypted files to each other.

S-HTTP will promote the growth of the electronic commerce as its transaction security features will promote a spontaneous commercial transactions. As S-HTTP allows Web clients and servers to be secured, the information exchanged among them will also be secured.

With S-HTTP, a secure server can reply to a request with a encrypted and signed message. By the same token, secure clients can verify a signature of a message and authenticate it. This authentication is done through the server's private key, which is used to generate the server's digital signatures. When the message was sent to the client, the server had delivered its public key certificate along with the signed message so that the client could verify the digital signature. The server can verify the integrity of a digitally-signed message send by a client through the same process of decrypting inbound messages from the client as well as encrypting outbound messages to the client.

You can encrypt data with shared, private or public keys. If data is encrypted with public keys, messages can be exchanged both ways and decrypted without the need for the client's public key as the server implement a single server private key that is stored in a key database file, which is encrypted using the Webmaster's password.

The encryption and signature is controlled through a CGI script. It is the local security configuration files and the CGI scripts S-HTTP message headers that will determine if the server will sign, encrypt, both or none.

Unfortunately, S-HTTP only works with SunOS 4.1.3, Solaris 2.4, Irix 5.2, HP-UX 9.03, DEC OSF/1, and AIX 3.2.4.

## ***Using SSL to Enhance Security***

The Secure Sockets Layer (SSL) protocol was designed and specified by Netscape Communications with the objective of improving data security layered between application protocols such as HTTP, TELNET, NNTP, FTP, and of course, TCP/IP.

SSL features data encryption, server authentication, message integrity, and optional client authentication for a TCP/IP connection.

This is an open, nonproprietary protocol, which was submitted by Netscape to the W3 Consortium for consideration as a standard security approach for Web browsers and servers. It has also been sent to the Internet Engineers Task Force (IETF) as an Internet Draft in the pursuit of having SSL standardized within the framework of the IETF.

SSL's main goal is to promote privacy and reliability between two communicating applications. The latest version, Version 3.0 of March 1996, supersedes the earlier version from December 1995.

Still, the bases of the protocol didn't change. It is a two layers protocol, relying, at the lowest level, on some reliable transport protocol, just like the HTTP protocol. This lower layer is called the SSL Record Protocol, which is used for encapsulation of various higher level protocols. One example is the SSL Handshake Protocol, which allows the server and the client to authenticate each other, as well as negotiate an encryption algorithm and cryptographic keys before any transmission.

The connection is private, the peer's identity can be authenticated using asymmetric or public key, and the connection

is reliable: this are the three basic properties of SSL.

The main difference between SSL and S-HTTP is that the later is a superset of the Web's HTTP, very specific to the Web usage. The SSL protocol, however, sends messages though a socket. The whole concept of SSL can be summarized in a protocol that can secure transactions between any client and server that use the sockets layer, which involves about all the TCP/IP application.

As far as encryption goes, both SSL and S-HTTP can negotiate different types of encryption algorithms and key authentication schemes, but Netscape and Enterprise Integration Technology (EIT) both have licensed RSA Data Security's toolkits to provide end-to-end encryption of messages, as well as key creation and certification.

Unfortunately, the future of electronic commerce and secure Web transaction can not rely in a multi-protocol security system. S-HTTP and SSL are not the same, nor work the same way. Fortunately, the Web Consortium is working hard to develop a unified security scheme that would include SSL and S-HTTP.

Moreover, these are not the only schemes been proposed. EInet's Secure Server, which uses Kerberos and other mechanisms, and the Shen proposal, suggest more comprehensive security than SSL or S-HTTP can offer, such as extensive use of Privacy-Enhanced Mail.

## ***Be Careful When Caching the Web!***

Caching can tremendously improve the performance of your Web service by ensuring that frequently requested files will tend to be stored in the local cache. However, if the file on the remote server is updated, an out-dated file will be retrieved from the cache by the user.

Also, those files can be retrieved by a remote user, revealing information that may not be for public or external users to read.

An HTTPD server can resolved this problem by looking at the date of the file on the remote server and comparing it with the one cached. The following is a typical cache log file recorded. It provides the domain name as well as the name of the machines:

```
xyz_pc77.leeds.ac.uk - - [21/Nov/1994:00:43:35 +0000] "GET
```

```
http://white.nosc.mil/gif_images/NM_Sunrise_s.gif HTTP/1.0" 200 18673
```

```
xyz_pc77.leeds.ac.uk - - [21/Nov/1994:00:43:38 +0000] "GET
```

```
http://white.nosc.mil/gif_images/glacier_s.gif HTTP/1.0" 200 6474
```

```
xyz_pc77.leeds.ac.uk - - [21/Nov/1994:00:43:40 +0000] "GET
```

```
http://white.nosc.mil/gif_images/rainier_s.gif HTTP/1.0" 200 18749
```

In the future it may be possible to chain caches. The possibility in the long term of having institutional, metropolitan, national and continental caches are beginning to be considered.

# ***Plugging the Holes: a Configuration Checklist***

Here are few configuration checklist to help you out:

- When configuring your HTTP server, never use raw IP addresses to allow access to your pages. Otherwise, you will end up with a bunch of them in your access list, which only will make maintenance harder.
- If you ever have problems with misconfigured clinet's domain server, have them contacting the LAN or systems administrator to fix it so you can reverse their names correctly. If you are the one to fix the problem, take the time and do it! In a the long run you will be thankful for it as otherwise, you may end up with a huge list of raw IP addresses on your list.
- You have to deal with access.conf files, make sure to put only one name per directive, which will ease the file editing, as you can comment out any directive by simply placing the "#" character at the start of the line.
- Remember to reboot your server after any changes made on your access.conf, as the changes you made will not take effect until you re-start the system.
- Always have an access control list of the top-level document directory. It will be useful when updating the file later.

## ***A Security Checklist***

First of all, the best security checklist you can have is knowing what to check and when. The following is a list of resources on the Internet to help you keeping abreast with security issues arising everyday in Cyberspace. It can also get some free resources to help you enhance security at your site:

- Subscribe to security mailing lists:
- Send an e-mail to the Computer Emergency Response Team (CERT) advisory mailing list, requesting your inclusion to their mailing list at [cert@cert.org](mailto:cert@cert.org).
- Try Phrack newsletter, an underground hacker's newsletter. Send an e-mail message to [phrack@well.sf.ca.us](mailto:phrack@well.sf.ca.us).
- Also try the Computer Underground Digest. Send e-mail to [tk0jut2@mvs.cso.niu.edu](mailto:tk0jut2@mvs.cso.niu.edu).

## ***Novell's HTTP: Better be Careful***

Novell's HTTP is known to have a very unsecured CGI script. If you are running a Novell Web server, you should disable the "convert.bas" CGI script it comes with.

Unfortunately, that script (the out of the box one!) allows any remote user to read any file on the remote server. How? Her it is the harmful code:

```
http://victim.com/scripts/convert.bas?../../../../anything/you/want/to/view
```

Novell will probably come up with a fix for this script, but as I write this chapter, to the best of my knowledge, no fixes have been provided. So make sure to disable the script when setting up you Novell HTTP!

## ***Watch for UNIX-based Web Server Security Problems***

History shows (see CERT's reports and Bulletin Advisories) that UNIX-based Web server have tendencies to breach security at:

- **Password Weakness** - Educate your user to pick passwords not found in dictionaries. Hackers often use finger or ruser to discover account names and then try to crack the password. A good heuristic for picking a password is to create a easy to remember phrase such as "Where is Carmen Sandiego" and then use the first letters of the words for password ("WICS"). Yet, try to choose passwords with at least 8 characters.

- **Unchanged Passwords** - Make sure to change default passwords when installing servers for the first time. Always remove unused accounts from the password file. Disable these account by changing the password field in the /etc/passwd file to an asterisk '\*' and change the login shell to /bin/false to ensure that an intruder cannot login to the account from a trusted system on the network.
- **Passwords Re-used** - Use passwords only once. Be aware that passwords can be captured over the Internet by sniffer programs.
- **Password Theft** - Hackers use Trivial File Transfer Protocol (TFTP) to steal password files. If you are not sure about this vulnerability at your system, connect to it using the TFTP protocol and try to get /etc/motd. If you can access it, then everyone on the Internet can get to your password file. To avoid it, either disable tftpd or restrict its access.

## URI/URL

Uniform Resource Identifiers (URI), are a group of extensive technologies for naming and addressing resources such as pages, services and documents on the web. There are a number of existing addressing schemes, and more may be incorporated over time.

Figure 4.1 shows the basic structure of URI which includes:

- **URI** - The Uniform Resource Identifier, a generic set of all names/addresses referring to resources.
- **URL** - The Uniform Resource Locator is a set of URI schemes with explicit instructions on how to access a resource on the Internet.
- **URN** - The Uniform Resource Name is composed of:
  - An URI that has an institutional commitment to persistence and availability, and a
  - A particular scheme, under development in the IETF to provide for the resolution using internet protocols of names which have a greater persistence than that currently associated with internet host names or organizations. When defined, a URN will be an example of a URI.
- **URC** - The Uniform Resource Citation, also known as Uniform Resource Characteristics, which is a set of attribute/value pairs describing a resource. These values could be URIs of various kinds, but it can also include, for example, authorship, publisher, data type, date, copyright status and so forth.

An Uniform Resource Locator (URL) is a sort of networked extension of the standard filename concept. An URL enables you to point to a specific file on a specific directory at any giving machine attached to the Internet or Intranet. Also, this file can be served though several different methods, such as HTTP, TELNET, FTP and so forth.

The following is an overview of some of the most common URL types, as described at the National Computer Security Associations' site at University of Illinois (<http://www.ncsa.uiuc.edu/demoweb/url-primer.html>).

## File URLs

Suppose there is a document called "foobar.txt"; it sits on an anonymous ftp server called "ftp.yoyodyne.com" in directory "/pub/files". The URL for this file is then:

```
file://ftp.yoyodyne.com/pub/files/foobar.txt
```

The top-level directory of this FTP server is simply:

```
file://ftp.yoyodyne.com/
```

The "pub" directory of this FTP server is then:

```
file://ftp.yoyodyne.com/pub
```

## ***Gopher URLs***

Gopher URLs are a little more complicated than file URLs, since Gopher servers are a little trickier to deal with than FTP servers. To visit a particular gopher server (say, the gopher server on gopher.yoyodyne.com), use this URL:

```
gopher://gopher.yoyodyne.com/
```

Some gopher servers may reside on unusual network ports on their host machines. (The default gopher port number is 70.) If you know that the gopher server on the machine "gopher.banzai.edu" is on port 1234 instead of port 70, then the corresponding URL would be:

```
gopher://gopher.banzai.edu:1234/
```

## ***News URLs***

To point to a Usenet newsgroup (say, "rec.gardening"), the URL is simply:

```
news:rec.gardening
```

## ***Partial URLs***

Once you are viewing a document located somewhere on the network (say, the document <http://www.yoyodyne.com/pub/afire.html>), you can use a partial, or relative, URL to point to another file in the same directory, on the same machine, being served by the same server software. For example, if another file exists in that same directory called "anotherfile.html", then anotherfile.html is a valid partial URL at that point.

This provides an easy way to build sets of hypertext documents. If a set of hypertext documents are sitting in a common directory, they can refer to one another (i.e., be hyperlinked) by just their filenames -- however a reader got to one of the documents, a jump can be made to any other document in the same directory by merely using the other document's filename as the partial URL at that point. The additional information (access method, hostname, port number, directory name, etc.) will be assumed based on the URL used to reach the first document.

## ***CGI***

Another form of threat that makes harder for a firewall to protect a Web site involves Common Gateway Interface (CGI) scripts. Many Web pages display documents and hyperlink them to other pages or sites. However, some have search engines that will allow you to search the site (or sites) for particular information. This is done through forms that are executed by CGI scripts.

Hackers can modify these CGI scripts to do things it really ought not to do. Normally, these CGI scripts will only search into the Web server area, but if you modify it, it can search outside the Web server. To prevent it from happening you will need to set these scripts with low user privileges, and if you are running a UNIX-based server, make sure you search for those semicolons again.



There are many known forms of threats and many more of unknown ones. In the next sections you learn about some of the most common and threatening ones.

Further, the open architecture of Web servers allows for arbitrary Common Gateway Interface (CGI) scripts to be executed on the server's side of the connection in response to remote requests. Any CGI script installed at your site may contain bugs, and every such bug is a potential security hole.

**Caution:**

Beware of CGI scripts, as they are the major source of security holes. The protocol itself is not insecure, but the scripts must be written with security in mind. If you are installing these scripts at your site, beware of the problem!

The same goes for Web server software, as more features they have greater is the potential for security holes. Servers that offer a variety of features such as CGI script execution, directory listing in real-time and script error handling, are more likely to be vulnerable to security holes. Even security tools widely used are not guaranteed to always work.

**Note:**

There is a Web server comparison table available at <http://www.webcompare.com/>. It includes freeware as well as commercial products for UNIX, Novell, Windows NT, Windows 95, VMS, and many other operating system.

For instance, right before I started writing this book, two present events come to mind. First, is about the well known Kerberos system, widely adopted for security in distributed systems, developed at MIT in the mid-1980s. The people from COAST, at Purdue University, found a vulnerability in current versions of the Kerberos. Couple students, Steve Lodin and Bryn Dole, and the professor Eugene Spafford, discovered a method where someone without privileged access to most implementations of a Kerberos 4 server could break secret session keys issued to users, allowing unauthorized access to distributed services available to a user without even knowing that user's password. They were able to demonstrate it in a record time of less than 1 minute, on average, using a typical workstation, and sometimes as quickly as 1/5 second!

Another example is Netscape, where versions 2.0 and 2.01 were vulnerable to a "malicious" Java applet being spread over the Internet, according to a story on the New York Times of May 18. This applet, although a bit annoying, could cause denial-of-service, which potentially could cause also loss of unsaved edits in a word processor, or erratic behavior of application if you, in a verge of panic decided to reboot your machine instead of just killing your browser.

**Note:**

What about Java?

Java is a language developed by Sun Microsystems which allows Web pages to contain codes to be executed by browsers. The exciting thing about Java is that, by being based on a single "virtual machine" that all implementations of Java emulates, it is capable to run on any system with a version of it. There is a web browser, HotJava, totally written in the Java language. If want to learn about it, try the URL: <http://java.sun.com>.

However, keep in mind that denial-of-service applets are not viruses, which are created with malicious intentions. True, this Java bug had the capability to execute instruction over the Web server, remotely, with the ability even to upload

information from within the remote Web server, but the security breaches that have gotten so much press were fixed in JDK 1.0.2, their current release, and in NN3.0b4.

In the interim, Netscape users were instructed to disable "Java" and "Java script" dialog box to prevent the browser from receiving such applets, or upgrade to version 2.02, which supposedly resolved the problem.

Another example you should be aware of is the existing vulnerability in the httpd servers provided by NCSA and the Apache organization. According to the Computer Incident Advisory Capability (CIAC), an user can potentially gain the same access privileges as the httpd server. This security hole not only applies to UNIX servers but to all server's platform capable of running httpd. If you are running an NCSA httpd, you should upgrade it to version 1.5.1, its latest version.

**Tip:**

You can download the NCSA httpd version 1.5 from the URL  
[ftp://ftp.ncsa.uiuc.edu/Web/httpd/UNIX/ncsa\\_httpd/current/httpd\\_1.5.1-export\\_source.tar.Z](ftp://ftp.ncsa.uiuc.edu/Web/httpd/UNIX/ncsa_httpd/current/httpd_1.5.1-export_source.tar.Z)

**Note:**

If you want to download patch 1.3 for NCSA's version 1.3 for UNIX, it is available at  
<http://hoohoo.ncsa.uiuc.edu/>.

The Apache plug-in replacement for NCSA can be found at  
<http://www.hyperreal.com/apache/info.html>).

The problem with the Apache httpd CGI is no different: a hacker could easily enter arbitrary commands on the server host using the same user-id as the user running the httpd server. If httpd is being run as root, the unauthorized commands are also run as root! Since he is using the same user-id, he can also access any file on the system that is accessible to the user-id that is running the httpd server, including but not limited to destroying file contents on the server host.

Further, if he is using an X11-based terminal emulator attached to the httpd server host, he can gain full interactive access to the server host just as if he were logging in locally.

If you are using Apache httpd, this is what you will need to do:

1. Locate the `escape_shell_command()` function in the file "src/util.c" (approximately line 430). In that function, the line should read `if(ind("&`\"|*?~<>^()[]{}$\\",cmd[x]) != -1){`
2. You will need to change that line to read `if(ind("&`\"|*?~<>^()[]{}$\\n",cmd[x]) != -1){`
3. Then, you will need to recompile, reinstall, and restart the server.

It is very important that you run the upgrade as if let alone, this security hole can lead to a compromise of your Web server.

**Note:**

For additional information you should visit CIAC's Web page at URL: <http://ciac.llnl.gov/>

The same goes for CGI scripts with Novell platforms. The challenge involved with the implementation of CGI gateways on Novell-based platforms is due to the overhead involved in spawning NLMs and implementing language

compilers or interpreters that reside and launch on the NetWare server. In order to resolve this problem, Great Lakes will allow data from the Web client to be either stored in a file on the NetWare server or transmitted as an MHS or SMTP E-mail message.

The NT version of both Netscape Communications Server version 1.12 and the Netscape Commerce Server, also are affected by CGI scripts handling. The following are two known problems:

- **Perl CGI Scripts are Insecure** - Since the Netscape server does not use the NT File Manager's associations between file extensions and applications, Perl scripts are not recognized as such when placed in to the cgi-bin directory. To associate the extension .pl with the Perl interpreter will not work. You are using any of these versions, Netscape technical note recommends to place Perl.exe into the cgi-bin and refer to your scripts as */cgi-bin/Perl.exe?&my\_script.pl*.

Unfortunately this technique opens a major security hole on the system as it allows a remote user to execute an arbitrary set of Perl commands on the server by invoking such scripts as */cgi-bin/Perl.exe?&-e+unlink+%3C\*%3E*, which will cause every file in the server's current directory to be removed.

There is another suggestion on Netscape's technical note to encapsulate the Perl scripts in a batch (.bat) file. However, be aware that there is also a related problem with batch scripts, which makes this solution unsafe.

Both Purveyor and WebSite NT servers, because of EMWACS, use NT's File Manager extension associations, allowing you to execute Perl scripts without having to place Perl.exe into cgi-bin. This bug does not affect these products.

- **DOS batch files are Insecure** - According to Ian Redfern (redferni@logica.com), a similar hole exists in the processing of CGI scripts implemented as batch files. Here it is how he describes the problem:

"Consider test.bat:

```
@echo off
```

```
echo Content-type: text/plain
```

```
echo
```

```
echo Hello World!
```

If you try to call it as */cgi-bin/test.bat?&dir* you will get the output of the CGI program, followed by a directory listing! It is like the server is executing two functions here, running the batch file test.bat and running a directory ('DIR' DOS Command) list, which the command interpreter is handling in the same way '/bin/sh' would (run it, then, if okay, run dir command).

A possible solution for this problem would be to wrap the batch file in to a compiled executable (.exe) file. The executable file would first checks the command line parameters for things that could be misinterpreted by DOS, then invoke a command.com subshell, and run the batch file.

This would require some extra work. You probably would be better off to do everything in compiled code. Again, if you are using this version, you definitely should upgrade it. You can easily do so by accessing Netscape's Web page at URL: <http://www.netscape.com>.

Also, keep in mind that there are several CGI script that allow users to change their passwords online. However, none of them have been tested enough to recommend. If you want to allow your users to change their passwords on-line, some sites have set up a second HTTP server for that sole purpose. This second server sort of replicate the password

file.

Further, if you have an FTP daemon, even though generally you would not be compromising data security by sharing directories between this daemon and your Web daemon, no remote user should ever be able to upload files that can later be read or executed by your Web daemon. Otherwise, a hacker could, for example, upload a CGI script to your ftp site and then use his browser to request the newly uploaded file from your Web server, which could execute the script, totally by-passing security! Therefore, limit ftp uploads to a directory that cannot be read by any user. More about this is discussed on chapter 8, "How Vulnerable are Internet Services."

Evidently, your Web servers should support the development of application gateways, as it is essential for communicating data between an information server--in this case a Web server--and another application.

Wherever the Web server needs to communicate with another application, you will need CGI scripts to negotiate the transactions between the server and an outside application. For instance, CGIs are used to transfer data, filled in by a user in an HTML form, from the Web server to a database.

But if you want to preserve the security of your site, and you must, be alert about allowing your users to run their own CGI scripts. These scripts are very powerful, which could represent some risks for your site. As discussed earlier, CGI scripts, if poorly written could open security roles in your system. Thus, never run your Web server as root; make sure it is configured to change to another user ID at startup time. Also, consider using a CGI wrapper to ensure the scripts run with the permissions and user id of the author. You can easily download one from URL:

<http://www.umn.edu/~cgiwrap>

### Tip:

You should check the URL: <http://www.primus.com/staff/paulp/cgi-security/> for security related scripts.

CGI are not all bad! A good security tool to control who is accessing your Web server is to actually use CGI scripts to identify them. There are five very important environment variables available to help you do that:

1. **HTTP\_FROM** - This variable is usually set to the email address of the user. You should use it as a default for the reply email address in an email form.
2. **REMOTE\_USER** - It is only set if secure authentication was used to access the script. You can use the AUTH\_TYPE variable to check what form of secure authentication was used. REMOTE\_USER will display the name of the user authenticated under.
3. **REMOTE\_IDENT** - It is set if the server has contacted an IDENTD server on the browser machine. However, there is no way to ensure a honest reply from the browser
4. **REMOTE\_HOST** - Provides information about the site the user is connecting from if the hostname was retrieved by the server.
5. **REMOTE\_ADDR** - This also provides information about the site the user is connecting from. It will provide the dotted-decimal IP address of the user.

### Caution:

If you ever suspect your site have been broken-in you should contact the Computer Emergency Response Team (CERT). CERT was formed by the Defense Advanced Research Projects Agency (DARPA) in 1988 to serve as a focal point for the computer security concerns of Internet users. The Software Engineering at Carnegie Mellon University, in Pittsburgh, PA runs the Coordination Center for the CERT. You can visit their Web page at URL: <http://www.cert.org> or send an e-mail to [cert@cert.org](mailto:cert@cert.org).

Also, CGI can be used to create e-mail forms on the Web. There is a CGI e-mail form, developed in Perl by Doug Stevenson (doug+@osu.edu), of Ohio State University, that is fairly secure. The script, called "Web Mailto Gateway," enables you to hide the real e-mail addresses from user, which helps to enhance security. The following source code can be found at URL: [http://www.mps.ohio-state.edu/mailto/mailto\\_info.html](http://www.mps.ohio-state.edu/mailto/mailto_info.html).

```
#!/usr/local/bin/perl

#

# Doug's WWW Mail Gateway 2.2

# 5/95

# All material here is Copyright 1995 Doug Stevenson.

#

# Use this script as a front end to mail in your HTML.  Not every browser

# supports the mailto: URLs, so this is the next best thing.  If you

# use this script, please leave credits to myself intact!  :)  You can

# modify it all you want, though.

#

# Documentation at:

#   http://www-bprc.mps.ohio-state.edu/mailto/mailto\_info.html

#

# Configurable items are just below.  Also pay special attention to

# GET method arguments that this script accepts to specify defaults
```

```
# for some fields.

#

# I didn't exactly follow the RFCs on mail headers when I wrote this,

# so please send all flames my way if it breaks your mail client!!

# Also, you'll need cgi-lib.pl for the GET and POST parsing.  I use

# version 1.7.

#

# Requires cgi-lib.pl which can be found at

#   http://www.bio.cam.ac.uk/web/form.html

#

# PLEASE: Use this script freely, but leave credits to myself!!  It's

#       common decency!

#

#####

#

# Changes from 1.1 to 1.2:

#

# A common modification to the script for others to make was to allow
```

```
# only a certain few mail addresses to be sent to.  I changed the WWW

# Mail Gateway to allow only those mail addresses in the list @addrs

# to be mailed to - they are placed in a HTML <SELECT> list, with either

# the selected option being either the first one or the one that matches

# the "to" CGI variable.  Thanks to Mathias Koerber

# <Mathias.Koerber@swi.com.sg> for this suggestion.

#

# Also made one minor fix.

#

#####

#

# Changes from 1.2 to 1.3:

#

# Enhancing the enhancements from 1.2.  You can now specify a real name

# or some kind of identifier to go with the real mail address.  This

# information gets put in the %addrs associative array, either explicitly

# defined, or read from a file.  Read the information HTML for instructions

# on how to set this up.  Also, real mail addresses may hidden from the
```

```
# user.  Undefine or set to zero the variable $expose_address below.

#

#####

#

# Changes from 1.3 to 1.4

#

# The next URL to be fetched after the mail is sent can be specified with

# the cgi varaible 'nexturl'.

#

# Fixed some stupid HTML mistake.

#

# Force user to enter something for the username on 'Your Email:' tag,

# if identd didn't get a username.

#

# Added Cc: field, only when %addrs is not being used.

#

#####

#
```



```
# Quickie patch to 1.41

#

# Added <PRE>formatted part to header entry to make it look nice and fixed a

# typo.

#

#####

#

# Version 2.0 changes

#

# ALL cgi varaibles (except those reserved for mail info) are logged

# at then end of the mail received.  You can put forms, hidden data,

# or whatever you want, and the info for each variable will get logged.

#

# Cleaned up a lot of spare code.

#

# IP addresses are now correctly logged instead of just hostnames.

#

# Made source retrieval optional.
```

#

#####

#

# Changes from 2.0 to 2.1

#

# Fixed stupid HTML error for an obscure case. Probably never noticed.

#

# Reported keys are no longer reported in an apparently random order; they

# are listed in the order they were received. That was a function of perl

# hashes...changed to a list operation instead.

#

#####

#

# Changes from 2.1 to 2.2

#

# Added all kinds of robust error checking and reporting. Should be

# easier to diagnose problems from the user end.

#

```
# New suggested sendmail flag -oi to keep sendmail from ending mail

# input on line containing . only.

#

# Added support for setting the "real" From address in the first line

# of the mail header using the -f sendmail switch.  This may or may not

# be what you want, depending on the application of the script.  This is

# useful for listservers that use that information for identification

# purposes or whatever.  This is NOT useful if you're concerned about

# the security of your script for public usage.  Your mileage will vary,

# please read the sendmail manual about the -f switch.

#   Thanks to Jeff Lawrence (jlaw@irus.rru.uwo.ca) for figuring this

#   one out.

#

#####

#

# Doug Stevenson

# doug+@osu.edu

#####
```

## # Configurable options

#####

# whether or not to actually allow mail to be sent -- for testing purposes

\$active = 1;

# Logging flag. Logs on POST method when mail is sent.

\$logging = 1;

\$logfile = '/usr/local/www/etc/mailto\_log';

# Physical script location. Define ONLY if you wish to make your version

# of this source code available with GET method and the suffix '?source'

# on the url.

\$script\_loc = '/usr/local/www/cgi-bin/mailto.pl';

# physical location of your cgi-lib.pl

\$cgi\_lib = '/usr/local/www/cgi-bin/cgi-lib.pl';

# http script location

\$script\_http = 'http://www-bprc.mps.ohio-state.edu/cgi-bin/mailto.pl';

# Path to sendmail and its flags. Use the first commented version and

# define \$listserver = 1 if you want the gateway to be used for listserver

# subscriptions -- the -f switch might be necessary to get this to work

```
# correctly.

#

# sendmail options:

#     -n    no aliasing

#     -t    read message for "To:"

#     -oi   don't terminate message on line containing '.' alone

#$sendmail = "/usr/lib/sendmail -t -n -oi -f";  $listserver = 1;

$sendmail = "/usr/lib/sendmail -t -n -oi";

# set to 1 if you want the real addresses to be exposed from %addrs

#$expose_address = 1;

# Uncomment one of the below chunks of code to implement restricted mail

# List of address to allow ONLY - gets put in a HTML SELECT type menu.

#

#%addrs = ("Doug - main address", "doug+@osu.edu",

           "Doug at BPRC", "doug@polarmet1.mps.ohio-state.edu",

           "Doug at CIS", "stevenso@cis.ohio-state.edu",

           "Doug at the calc lab", "dstevens@mathserver.mps.ohio-state.edu",

           "Doug at Magnus", "dmsteven@magnus.acs.ohio-state.edu");
```

```

# If you don't want the actual mail addresses to be visible by people
# who view source, or you don't want to mess with the source, read them
# from $mailto_addrs:

#

# $mailto_addrs = '/usr/local/www/etc/mailto_addrs';

# open(ADDRS, $mailto_addrs);

# while(<ADDRS>) {

#     ($name, $address) = /^(.+)[ \t]+([^\s]+)\n$/;

#     $name =~ s/[ \t]*$//;

#     $addrs{$name} = $address;

# }

# version

$version = '2.2';

#####

# end of configurable options

#####

#####

# source is self-contained

```

#####

```

if ($ENV{'QUERY_STRING'} eq 'source' && defined($script_loc)) {

    print "Content-Type: text/plain\n\n";

    open(SOURCE, $script_loc) ||

        &InternalError('Could not open file containing source code');

    print <SOURCE>;

    close(SOURCE);

    exit(0);

}

require $cgi_lib;

&ReadParse();

#####

# method GET implies that we want to be given a FORM to fill out for mail

#####

if ($ENV{'REQUEST_METHOD'} eq 'GET') {

    # try to get as much info as possible for fields

    # To:      comes from $in{'to'}

    # Cc:      comes from $in{'cc'}

```

```
# From:    comes from REMOTE_IDENT@REMOTE_HOST || ${in{'from'}} || REMOTE_USER

# Subject: comes from ${in{'sub'}}

# body comes from ${in{'body'}}

$destaddr = ${in{'to'}};

$cc = ${in{'cc'}};

$subject = ${in{'sub'}};

$body = ${in{'body'}};

$nexturl = ${in{'nexturl'}};

if (${in{'from'}}) {

    $fromaddr = ${in{'from'}};

}

# this is for NetScape pre-1.0 beta users - probably obsolete code

elseif ($ENV{'REMOTE_USER'}) {

    $fromaddr = $ENV{'REMOTE_USER'};

}

# this is for Lynx users, or any HTTP/1.0 client giving From header info

elseif ($ENV{'HTTP_FROM'}) {

    $fromaddr = $ENV{'HTTP_FROM'};
```



```

}

# if all else fails, make a guess

else {

    $fromaddr = "$ENV{'REMOTE_IDENT'}\@$ENV{'REMOTE_HOST'}";

}

# Convert multiple bodies (separated by \0 according to CGI spec)

# into one big body

$body =~ s/\0//;

# Make a list of authorized addresses if %addrs exists.

if (%addrs) {

    $selections = '<SELECT NAME="to">';

    foreach (sort keys %addrs) {

        if ($in{'to'} eq $addrs{$_}) {

            $selections .= "<OPTION SELECTED>$_";

        }

        else {

            $selections .= "<OPTION>$_";

        }

    }

}

```

```

        if ($expose_address) {

            $selections .= " <$addr{$_}>";

        }
    }

    $selections .= "</SELECT>\n";

}

```

```

# give them the form

```

```

print &PrintHeader();

```

```

print <<EOH;

```

```

<HTML><HEAD><TITLE>Doug\'s WWW Mail Gateway $version</TITLE></HEAD>

```

```

<BODY><H1><IMG SRC="http://www-bprc.mps.ohio-state.edu/pics/mail2.gif" ALT="">

```

```

The WWW Mail Gateway $version</H1>

```

```

<P>The <B>To</B>: field should contain the <B>full</B> Email address

```

```

that you want to mail to. The <B>Your Email</B>: field needs to

```

```

contain your mail address so replies go to the right place. Type your

```

```

message into the text area below. If the <B>To</B>: field is invalid,

```

```

or the mail bounces for some reason, you will receive notification

```

```

if <B>Your Email</B>: is set correctly. <I>If <B>Your Email</B>:

```

is set incorrectly, all bounced mail will be sent to the bit bucket.</I></P>

```
<FORM ACTION="$script_http" METHOD=POST>
```

```
EOH
```

```
;
```

```
print "<P><PRE>          <B>To</B>: ";
```

```
# give the selections if set, or INPUT if not
```

```
if ($selections) {
```

```
    print $selections;
```

```
}
```

```
else {
```

```
    print "<INPUT VALUE=\"\$destaddr\" SIZE=40 NAME=\"to\">\n";
```

```
    print "          <B>Cc</B>: <INPUT VALUE=\"\$cc\" SIZE=40 NAME=\"cc\">\n";
```

```
}
```

```
print <<EOH;
```

```
<B>Your Name</B>: <INPUT VALUE="$fromname" SIZE=40 NAME="name">
```

```
<B>Your Email</B>: <INPUT VALUE="$fromaddr" SIZE=40 NAME="from">
```

```
<B>Subject</B>: <INPUT VALUE="$subject" SIZE=40 NAME="sub"></PRE>
```

```
<INPUT TYPE="submit" VALUE="Send the mail">
```

<INPUT TYPE="reset" VALUE="Start over"><BR>

<TEXTAREA ROWS=20 COLS=60 NAME="body">\$body</TEXTAREA><BR>

<INPUT TYPE="submit" VALUE="Send the mail">

<INPUT TYPE="reset" VALUE="Start over"><BR>

<INPUT TYPE="hidden" NAME="nexturl" VALUE="\$nexturl"></P>

</FORM>

<HR>

<H2>Information about the WWW Mail Gateway</H2>

<H3><A HREF="http://www-bprc.mps.ohio-state.edu/mailto/mailto\_info.html#about">

About the WWW Mail Gateway</A></H3>

<H3><A HREF="http://www-bprc.mps.ohio-state.edu/mailto/mailto\_info.html#new">

New in version \$version</A></H3>

<H3><A HREF="http://www-bprc.mps.ohio-state.edu/mailto/mailto\_info.html#misuse">

Please report misuse!</A></H3>

<HR>

<ADDRESS><P><A HREF="/~doug/">Doug Stevenson: doug+\@osu.edu</A>

</P></ADDRESS>

</BODY></HTML>

EOH

;

}

#####

# Method POST implies that they already filled out the form and submitted

# it, and now it is to be processed.

#####

elif (\$ENV{'REQUEST\_METHOD'} eq 'POST') {

# get all the variables in their respective places

\$destaddr = \$in{'to'};

\$cc = \$in{'cc'};

\$fromaddr = \$in{'from'};

\$fromname = \$in{'name'};

\$replyto = \$in{'from'};

\$sender = \$in{'from'};

\$errorsto = \$in{'from'};

\$subject = \$in{'sub'};

\$body = \$in{'body'};

```
$nexturl = $in{'nexturl'};
```

```
$realfrom = $ENV{'REMOTE_HOST'} ? $ENV{'REMOTE_HOST'}: $ENV{'REMOTE_ADDR'};
```

```
# check to see if required inputs were filled - error if not
```

```
unless ($destaddr && $fromaddr && $body && ($fromaddr =~ /^.+\@.+\/)) {
```

```
    print <<EOH;
```

Content-type: text/html

Status: 400 Bad Request

<HTML><HEAD><TITLE>Mailto error</TITLE></HEAD>

<BODY><H1>Mailto error</H1>

<P>One or more of the following necessary pieces of information was missing  
from your mail submission:

<UL>

<LI><B>To</B>:, the full mail address you wish to send mail to</LI>

<LI><B>Your Email</B>: your full email address</LI>

<LI><B>Body</B>: the text you wish to send</LI>

</UL>

Please go back and fill in the missing information.</P></BODY></HTML>

EOH

```
exit(0);

}

# do some quick logging - you may opt to have more/different info written

if ($logging) {

    open(MAILLOG, ">>$logfile");

    print MAILLOG "$realfrom\n";

    close(MAILLOG);

}

# Log every CGI variable except for the ones reserved for mail info.

# Valid vars go into @data.  Text output goes into $data and gets.

# appended to the end of the mail.

# First, get an ORDERED list of all cgi vars from @in to @keys

for (0 .. $#in) {

    local($key) = split(/=/,$in[$_],2);

    $key =~ s/\+/ /g;

    $key =~ s/%(..)/pack("c",hex($1))/ge;

    push(@keys,$key);

}
```

```
# Now weed out the ones we want
```

```
@reserved = ('to', 'cc', 'from', 'name', 'sub', 'body', 'nexturl');
```

```
local(%mark);
```

```
foreach (@reserved) { $mark{$_} = 1; }
```

```
@data = grep(!$mark{$_}, @keys);
```

```
foreach (@data) {
```

```
    $data .= "$_ -> $in{$_}\n";
```

```
}
```

```
# Convert multiple bodies (separated by \0 according to CGI spec)
```

```
# into one big body
```

```
$body =~ s/\0//;
```

```
# now check to see if some joker changed the HTML to allow other
```

```
# mail addresses besides the ones in %addrs, if applicable
```

```
if (%addrs) {
```

```
    if (!scalar(grep($_." <$addrs{$_}>" eq $destaddr ||
```

```
        $destaddr eq $_, keys(%addrs)))) {
```

```
        print &PrintHeader();
```

```
        print <<EOH;
```



```
<HTML><HEAD><TITLE>WWW Mail Gateway: Mail address not allowed</TITLE></HEAD>
```

```
<BODY>
```

```
<H1>Mail address not allowed</H1>
```

```
<P>The mail address you managed to submit, <B>$destaddr</B>, to this script is  
not one of the pre-defined set of addresses that are allowed. Go back and  
try again.</P>
```

```
</BODY></HTML>
```

```
EOH
```

```
;
```

```
    exit(0);
```

```
}
```

```
}
```

```
# if we just received an alias, then convert that to an address
```

```
$realaddr = $destaddr;
```

```
if ($addrs{$destaddr}) {
```

```
    $realaddr = "$destaddr <$addrs{$destaddr}>";
```

```
}
```

```
# fork over the mail to sendmail and be done with it
```

```
if ($active) {

    if ($listserver) {

        open(MAIL,"| $sendmail$fromaddr") ||

            &InternalError('Could not fork sendmail with -f switch');

    }

    else {

        open(MAIL,"| $sendmail") ||

            &InternalError('Could not fork sendmail with -f switch');

    }

    # only print Cc if we got one

    print MAIL "Cc: $cc\n" if $cc;

    print MAIL <<EOM;

From: $fromname <$fromaddr>

To: $realaddr

Reply-To: $replyto

Errors-To: $errorsto

Sender: $sender

Subject: $subject
```

X-Mail-Gateway: Doug\'s WWW Mail Gateway \$version

X-Real-Host-From: \$realfrom

\$body

\$data

EOM

```
close(MAIL);
```

```
}
```

```
# give some short confirmation results
```

```
#
```

```
# if the cgi var 'nexturl' is given, give out the location, and let
```

```
# the browser do the work.
```

```
if ($nexturl) {
```

```
    print "Location: $nexturl\n\n";
```

```
}
```

```
# otherwise, give them the standard form.
```

```
else {
```

```
    print &PrintHeader();
```

```
    print <<EOH;
```

```
<HTML><HEAD><TITLE>Mailto results</TITLE></HEAD>

<BODY><H1>Mailto results</H1>

<P>Mail sent to <B>$destaddr</B>:<BR><BR></P>

<PRE>

<B>Subject</B>: $subject

<B>From</B>: $fromname &lt;$fromaddr>

$body</PRE>

<HR>

<A HREF="$script_http">Back to the WWW Mailto Gateway</A>

</BODY></HTML>
```

EOH

```
    ;

    }

}                                # end if METHOD=POST
```

#####

# What the heck are we doing here????

#####

else {

```
print <<EOH;
```

```
<HTML><HEAD><TITLE>Mailto Gateway error</TITLE></HEAD>
```

```
<BODY><H1>Mailto Gateway error</H1>
```

```
<P>Somehow your browser generated a non POST/GET request method and it  
got here. You should get this fixed!!</P></BODY></HTML>
```

```
EOH
```

```
}
```

```
exit(0);
```

```
#
```

```
# Deal out error messages to the user. Gets passed a string containing
```

```
# a description of the error
```

```
#
```

```
sub InternalError {
```

```
    local($errmsg) = @_;
```

```
    print &PrintHeader();
```

```
    print <<EOH;
```

```
Content-type: text/html
```

```
Status: 502 Bad Gateway
```

<HTML><HEAD><TITLE>Mailto Gateway Internal Error</TITLE></HEAD>

<BODY><H1>Mailto Gateway Internal Error</H1>

<P>Your mail failed to send for the following reason:<BR><BR>

<B>\$errmsg</B></P></BODY></HTML>

EOH

```
exit(0);
```

```
}
```

```
##
```

```
## end of mailto.pl
```

```
##
```

If your server can run CGI scripts and is configured with sendmail, this is the right, and secure, mail gateway script to have in your HTML, you will need to be able to run CGI scripts on your server though.

|          |          |
|----------|----------|
| Orders   | Orders   |
| Comments | Comments |

|      |         |
|------|---------|
| Back | Forward |
|------|---------|

---

[COMPUTING MCGRAW-HILL](#) | [Beta Books](#) | [Contact Us](#) | [Order Information](#) | [Online Catalog](#)

---



HTML conversions by [Mega Space](#).

This page updated on December 05, 1997 by [Webmaster](#).

[Computing McGraw-Hill](#) is an imprint of the [McGraw-Hill Professional Book Group](#).

Copyright ©1997 [The McGraw-Hill Companies, Inc.](#) All Rights Reserved.

Any use is subject to the rules stated in the [Terms of Use](#).

|        |        |
|--------|--------|
| Orders | Orders |
|--------|--------|

|      |         |
|------|---------|
| Back | Forward |
|------|---------|

|          |          |
|----------|----------|
| Comments | Comments |
|----------|----------|

© 1997 The McGraw-Hill Companies, Inc. All rights reserved.  
Any use of this Beta Book is subject to the rules stated in the [Terms of Use](#).

---

## Chapter 5

# ***Firewalling Challenges: The Advanced Web***

For the most part, Internet managers are used to the idea that a proxy server, a specialized HTTP server typically running on a firewall machine, would be enough to provide secure access from Internet connections coming through the firewall into the protected network.

Sure enough, running a proxy server is one of the most recommended approach to protect your Web site. But there is more to it than only setting up a proxy, which many times can breach security requirements. Thus, SOCKS comes in the picture. As a package that enables Internet clients to access protected network without breaching security requirements, SOCKS can also be an add-on feature to your firewall challenge. But not so fast! According to Ying-Da Lee (ylee@syl.dl.nec.com), from NEC, you may bump into few problems using the modified version of Mosaic for X 2.0, which is not supported by its developer, the National Computer Security Association (NCSA).

Therefore, to implement security in a Web environment is not really the same as to building an Internet firewall. To better understand the challenges in setting up a firewall in a Web-centric environment you must understand the threats and risks you are up against, as well as the implications of integrating different technologies, which includes but are not limited to protocols, devices and services.

This chapter discusses the main security flaws and risks associated to Web-based connectivity, as well as of the main technologies interacting with the Web, such as media types, programming languages and other security concerns, so that you can better choose and implement the right firewall solution.

## ***Extending the Web Server: Increased Risks***

As information technologies becomes a commodity to the whole Cyberspace, than everyone wants to have access to it, to use it and... to abuse it. It becomes an instrument of value, like any other commodity. Thus, it must be protected before it is stolen.

Unfortunately, there is a mob of talented hackers and crackers out there, a mix of cyberpunks and

whackers, lurking around, waiting for an opportunity to break-in to a secure system, regardless if it is a Web site or a corporate internal network. They will try to exploit anything, from high level Application Programming Interface (API) to low level services, from malicious applets, to sophisticated client-pull and server-push schemes.

What are they after? You should expect them to be after anything! Many of them will try the same old tricks UNIX crackers did years ago just for the fun of it. What about publicly posting your client list on the Internet? What if suddenly, instead of your company's logo you find one of those looney tunes character on your home page? Worse! What if you are been hacked right now and not even noticed? One thing you can be sure: soon or later they will knock your door, it is just a matter of statistics!

The bottom line is there always will be Web security issues you should be concerned with. Many of this security issues are documented at <http://www-genome.wi.mit.edu/WWW/faqs/www-security-faq.html>, at least for UNIX boxes. Therefore, lets take a look at some of the ways your Web server can be attacked and what you can do to prevent it.

## ISAPI

When time for integration between systems comes you will need to decide on the approach you will use to create interaction between your applications and the your Webserver. If you don't have an Intranet already in place, don't worry, you will! But before even considering it, you will first need to consider how your users will interact with the system you have in place and decide their level of interaction with your Web-centric applications.

The choice you make largely depends on what user interactivity you would like to build into the system. Some aspects of this interactivity are new, and some have been a part of LAN connectivity for some time. Ideally, when your application is linked with a Web server, your users will be able to use your application in ways unique to being on a Web, whether it is an "Intranet" or the Internet itself.

Be careful when choosing your Web server though. My recommendation goes for the Purveyor WebServer (<http://www.process.com>), which has much to offer your existing application and user base. For instance, Purveyor allows you to use existing user authentication and authorization systems or take advantage of user authentication and authorization using Purveyor. LAN-based applications can also use Purveyor's encryption services if desired. Also, since Purveyor can be configured as a proxy server, it may also be used to allow secure Internet access for users on the LAN. You may also want to consider the added user interactivity unique to Web technology.

The reason am highlighting this is that, by considering these design elements beforehand will save you programming time. Regardless of your Web server, depending on what you wish to do, you may not have much options when choosing how to access server functions, which will be by either of two major interfaces: the Common Gateway Interface (CGI) or the Internet Server Application Programming Interface (ISAPI). CGI provides a versatile interface that is portable between systems. ISAPI is much faster but requires that you write a Windows DLL, which is not a trivial programming exercise.

In all considered, the Internet Server Application Programming Interface, (ISAPI) is a high performance interface to back end applications running on your web server. Based on its own DLL ensuring significant performance over CGI, ISAPI is easy to use, well documented, and does not require complex



programming. These approaches are often combined. Some parts of your interface program may call DLLs and others may use the CGI approach. So let's take a look at the CGI approach first then the ISAPI one, so you can have a clear idea of what's involved as far as security.

**Note:**

Just for the record, if you're interested in CGI scripts, a good CGI tutorial can be found at URL <http://hoohoo.ncsa.uiuc.edu/cgi/>.

## CGI

The Common Gateway Interface is a standard method for writing programs to work with World Wide Web servers. Programs that use the Common Gateway interface, referred to as CGI scripts, usually take input from HTML forms to execute particular tasks. Developers may find it appropriate to use CGI in cases where ease of development and portability to other operating systems are important. CGI scripts are simple to write, and since the user interface is HTML, the CGI script can be initiated by any client that can run a browser.

As you know, users interact with Web Servers by filling in and submitting HTML forms or clicking on links in HTML documents. Through these HTML forms or links, the Web can be used to obtain important information and perform specific tasks. Routine tasks can be moved on-line, facilitating collaboration on projects between individuals and groups. HTML forms can also allow users to specify what information they want to obtain and what tasks they want to perform.

A CGI script can be an individual executable program or a chain of programs that can be started by the Purveyor Server in response to a client request. A typical CGI script may, for instance, take a keyword that a user has submitted in an HTML form and search for that keyword in a specific document or group of documents. When a user enters this keyword and submits it, the server passes this data to the CGI script. This program performs operations with the data, sending it back or passing it along to other applications as specified. When the data finally returns to the server, it is re-formatted into HTML and shipped back to the requesting client. Figure 5.1 illustrates this process.

However, CGIs have their limitations. In designing CGI scripts, bear in mind that each time the Web server executes a script it creates a new process and a new drain on available resources. This is one of the less attractive characteristics of the CGI method. It requires the server to spawn a new process every time a client invokes a CGI script. Each CGI call therefore consumes CPU time and server resources so that many simultaneous requests slow the entire system significantly. This problem can become particularly serious on a busy server with many concurrent requests. Consequently, the more calls there are to an application, the less suited it may be to CGI scripting because of the load this places on the server.

Bear in mind also that applications that use the power of corporate and business-to-business intranets often experience many more "hits" per hour than even the most popular internet Web sites.

Furthermore, CGI programs work within the constraints of the HTTP server. They communicate with the user through a stateless protocol, so they "forget" every previous transaction. There is no way of creating intensely interactive applications unless you arrange each step to re-transmit any information that has to

be "remembered" from previous steps. Although it is possible to write a program or a group of programs that build on previous information, you must write them with this stateless environment in mind.

## Internet Server API (ISAPI)

For cases where peak efficiency is more important than portability to other systems, the best method for extending Web server functionality is by using the Internet Server Application Programming Interface (ISAPI). Applications using ISAPI are compiled into Dynamic Link Library files (DLLs) that the Web server loads at startup. ISAPI programs have several key advantages over CGI scripts:

They are more efficient than CGI scripts because each client request does not spawn a new process.

Because ISAPI applications are more efficient than CGI scripts and are loaded into memory when the server starts, their performance is substantially superior to CGI programs.

These executables are the "native" method for extended functionality in the Windows environment. For example, the Microsoft Win32 Application Programming Interface is a set of Dynamic Link Libraries.

ISAPI was jointly developed by Process Software and Microsoft Corporation. It has been offered as a standard for all operating systems that support sharable images. It is an open specification. We have used it for Windows NT, Windows 95, NetWare and OpenVMS systems. Microsoft uses it on its Internet Information Server (IIS).

ISAPI applications run by making calls to resource files called Dynamic Link Libraries. Dynamic Link Libraries (or DLLs), are executable modules containing functions that applications can call to perform useful tasks. ISAPI DLLs exist primarily to provide services for Web application modules. These DLLs are referred to as *extension DLLs*.

Extension DLLs have a number of technical advantages:

- Several applications can share a single copy of any library function within a DLL.
- Extension DLLs load into the server's process space—eliminating the time and resource demands of creating additional processes.
- All resources available to the server are also available to its DLLs.
- DLLs execute with minimal overhead—considerably faster than EXE files.

In addition, server can manage DLLs, pre-loading commonly-used ones and unloading those that remain unused for some (configurable) period of time. The primary disadvantage in using an Extension DLL is that a DLL crash can cause a server crash.

These advantages make ISAPI an ideal interface for supporting server applications subject to heavy traffic in corporate intranets. As a matter of fact, the greater the degree of interactivity required of an Web server application, the more the application may be suited to an ISAPI interface. For example, engineers at Process Software use the ISAPI method to support the Purveyor Web Server's remote server management (RSM) application for just this reason. A sample screen from the RSM application is shown on figure 5.2.

The particular method used for ISAPI is called *run-time dynamic linking*. In this method, an existing program uses the **LoadLibrary** and **GetProcAddress** functions to get the starting address of DLL

functions, calls them through a common entry point called `HttpExtensionProc()`, and communicates with them through a data structure called an *Extension Control Block*.

The other method is called *load-time dynamic linking*, which requires building the executable module of the main application (the server) while linking with the DLL's import library. This method is not suitable for our purposes since it presents barriers to efficient server management of DLL applications.

How does the server handle the DLLs? The filename extension "DLL" in client requests is reserved for Dynamic Link Library files to be used through this Application Programming Interface. All extension DLLs must be named in the form \*.DLL and no other type of Purveyor Server executables requested by a client may have names of this form.

When the server gets a request to execute a DLL file, it takes the following steps:

- Checks to see if the requested DLL is already in memory and load it if not already present. If the DLL does not contain the entry point **GetExtensionVersion**, the server will not load it.
- Executes a call to the entry point **GetExtensionVersion** to verify that this DLL was written to conform to the API standard. If the returned value is not valid the server unloads the DLL without executing it.
- Executes a call to **HttpExtensionProc** to begin execution of the DLL.
- Responds as needed to the running DLL through the callback functions and the Extension Control Block.
- Terminates the operation upon receipt of a return value. If there is a non-null log string, the server writes the DLL's log entry to its log.

All Extension DLLs must export two entry points:

- **GetExtensionVersion()** - the version of the API specification to which the DLL conforms.

This entry point is used as a check that the DLL was actually designed to meet this specification, and specifies which version of this specification it uses. As additional refinements take place in the future, there may be additions and changes which would make the specification number significant. Table 5.1 shows a sample of a suitable definition in C.

- **HttpExtensionProc()** - the entry point for execution of the DLL. This entry point is similar to a `main()` function in a script executable. It is the actual startup of the function and has a form (coded in C) as described on Table 5.2.

**Table 5.1** - Using `GetExtensionVersion()` as an entry point.

```

BOOL WINAPI GetExtensionVersion( HSE_VERSION_INFO *version )
{
    version->dwExtensionVersion = HSE_VERSION_MAJOR;
    version->dwExtensionVersion = version->dwExtensionVersion << 16;
    version->dwExtensionVersion = version->dwExtensionVersion | HSE_VERSION_MINOR;
    sprintf( version->lpszExtensionDesc, "%s", "This is a sample Extension DLL" );

```

```
return TRUE;
```

**Table 5.2** - Using HttpExtensionProc() as an entry point.

```
DWORD WINAPI HttpExtensionProc(LPEXTENSION_CONTROL_BLOCK lpEcb);
```

Upon termination, ISAPI programs must return one of the following codes:

### **HSE\_STATUS\_SUCCESS**

The Extension DLL has finished processing and the server can disconnect and free up allocated resources.

### **HSE\_STATUS\_SUCCESS\_AND\_KEEP\_CONN**

The Extension DLL has finished processing and the server should wait for the next HTTP request if the client supports persistent connections. The Extension should only return this if they were able to send the correct Content-Length header to the client.

### **HSE\_STATUS\_PENDING**

The Extension DLL has queued the request for processing and will notify the server when it has finished (see HSE\_REQ\_DONE\_WITH\_SESSION under the Callback Function ServerSupportFunction ).

### **HSE\_STATUS\_ERROR**

The Extension DLL has encountered an error while processing the request and the server can disconnect and free up allocated resources.

There are four Callback Functions used by DLLs under this specification:

- **GetServerVariable** - obtains information about a connection or about the server itself. The function copies information (including CGI variables) relating to an HTTP connection or the server into a buffer supplied by the caller. If the requested information pertains to a connection, the first parameter is a connection handle. If the requested information pertains to the server, the first parameter may be any value except NULL.
- **ReadClient** - reads data from the body of the client's HTTP request. It reads information from the body of the Web client's HTTP request into the buffer supplied by the caller. Thus, the call might be used to read data from an HTML form which uses the POST method. If more than \*lpdwSize bytes are immediately available to be read, ReadClient will return after transferring that amount of data into the buffer. Otherwise, it will block waiting for data to become available. If the client's socket is closed, it will return TRUE but with zero bytes read.
- **WriteClient** - write data to the client. This function sends information to the Web client from the buffer supplied by the caller.

- **ServerSupportFunction** - provides the Extension DLLs with some general purpose functions as well as functions that are specific to HTTP server implementation. This function sends a service request to the server.

The server calls your application DLL at `HttpExtensionProc()` and passes it a pointer to the ECB structure. Your application DLL then decides what exactly needs to be done by reading all the client input (by calling the function **GetServerVariable()** ). This is similar to setting up environment variables in a Direct CGI application.

Since the DLL is loaded into the same process address space as that of HTTP server, an access violation by the Extension DLL crashes the server application. Ensure the integrity of your DLL by testing it thoroughly. DLL errors can also corrupt the server's memory space or may result in memory or resource leaks. To take care of this problem, a server should wrap the Extension DLL entry point in a "try/except clause" so that access violations or other exceptions will not directly effect the server. For more information on the "try/except" clause, refer to the help section on C/C++ Language under Visual C++ v2.0 help.

Although it may initially require more development resources to write the DLLs needed to run ISAPI applications, the advantages of using ISAPI are evident. ISAPI makes better use of system resources by keeping shared functions in a single library, and spawning only a single process for applications invoked by more than one client. The fact that the server pre-loads these libraries at startup ensures quicker program performance and faster server response time. Finally, the quickness and efficiency of ISAPI make it well suited for applications that require user interaction and that may be subject to heavy traffic, such as those that take full advantage of the intranet.

**Note:**

For more information on ISAPI programming, you may wish to participate in the Microsoft forum - ISAPI-L. You can subscribe by sending e-mail to:

`LISTSERV@peach.ease.lsoft.com`.

Include a one-line message with the body:

`SUBSCRIBE ISAPI-L <firstname><lastname>`

To send messages to the mailing list, e-mail them to:

`ISAPI-L@peach.ease.lsoft.com`

Microsoft has also made several PowerPoint presentations that deal with ISAPI development available at the following URL: <http://www.microsoft.com/intdev/pdc/pdcserv.htm>. These presentations describe ISAPI advantages, filters and programming techniques while providing several examples of ISAPI applications.

**Tip:**

For your information, ISAPI is available on Purveyor for Windows NT and Purveyor for OpenVMS, developed by Process Software Corp. For additional information and program documentation, check their Web site at URL <http://www.process.com/news/spec.htm>.

## A Security Hole on IIS exploits ISAPI

However, if Web developers take advantage of ISAPI wonderful features, so do hackers, by reverting the "IUSR\_MACHINENAME" account" of Microsoft's Internet Information Server (IIS).

The exploit here is that ISAPI scripts run under the IUSR\_MACHINENAME account under IIS, so ISAPI inherits the security permissions of this account. Thus, if the ISAPI program were to contain a simple call labeled "RevertToSelf()," for example, there you have a major hole!

As soon as that line of the program is executed, the ISAPI program reverts its authority to the system account, which hold all access privileges on the server. At this point, a hacker is capable to execute anything on the server, including "system()" calls.

**Caution:**

If you want to try the exploit above, check the URL <http://www.ntsecurity.net/security/webguest.htm>, which has a DLL called REVERT.DLL that you can run from any Intel based IIS box. The script, once downloaded to your scripts directory on the IIS machine, once executed, will create a directory called C:\IIS-REVERT-TEST without your authorization!

## What can you do About it?

Not much can be done to prevent this exploit. Don't be naive, don't run any ISAPI scripts that you don't understand or don't trust the source code, especially it comes from a shareware or freeware site! A good measure is to compile the source code yourself. I would not recommend you to run a script without compiling it or trusting the developer/source code.

Also, make sure to test the ISAPI applications as much as you can on a standalone machine before you make it available on the Net.

**Tip:**

To learn more about ISAPI, check the ISAPI Tutorials page at URL <http://www.genusa.com/isapi/isapitut.htm>

# NSAPI

Netscape Server Application Programming Interface (NSAPI) is Netscape's version of ISAPI, which also works on UNIX systems that support shared objects, and can be used as a framework for implementing custom facilities and mechanisms. However, NSAPI groups a series of functions to be used specifically with Netscape Server, allowing it to extend the core functionality of the Netscape Server. According to Netscape (<http://developer.netscape.com/misc/developer/conference/proceedings/s5/sld002.html> ) NSAPI provides flexibility, control, efficiency, and multi-platform solutions which includes but is not limited to:

- Faster CGI-type functions
- Database connectivity
- Customized logging
- Version control
- Personalized web site for each clients
- Alternative access control
- Custom user authentication
- Revised version of an existing server functionality
- Plug-in applications

The Yale University suggests NSAPI to be very efficient (<http://pclt.cis.yale.edu/pclt/webapp/apis.htm>). This is easy to grasp as NSAPI works very tight with the Netscape Server. The functions that Netscape provides through the NSAPI interface can locate information and

[Orders](#)[Orders](#)[Back](#)[Forward](#)[Comments](#)[Comments](#)

© 1997 The McGraw-Hill Companies, Inc. All rights reserved.  
Any use of this Beta Book is subject to the rules stated in the [Terms of Use](#).

---

## Chapter 6

# *The APIs Security Holes and Its Firewall Interactions*

An Application Program Interface (API) is the specific method prescribed by a computer operating system by which a programmer writing an application program can make requests of the operating system.

An API can be contrasted with an interactive user interface or a command interface as an interface to an operating system. All of these interfaces are essentially requests for system services.

As discussed on chapter 5, "Firewalling Challenges: the Advanced Web," under the sections on ISAPI and NSAPI, API provides another alternative to CGI, SSI and Server-Side Scripting for working with Web servers, creating dynamic documents, as well as providing other services via the Web.

However, I believe that for the most part, you should try to develop Web-centric application not only with APIs, but also using SSI, CGI and SSS technology, which is discussed in more details on chapter 5. The reason I say this is because I also believe there has been too much media hype lately about pseudo-standard API technology. Much of it is about its speed when compared to CGI scripts, but this information overlooks some vital facts: Your choice of webserver should be heavily influenced by its SSI, SSS, and CGI capabilities and efficiency as well as its support for advanced API programming. Otherwise, you gain nothing. O'Reilly has a great paper at their Web site (<http://website.ora.com/devcorner/white/extending.html>) which discusses this issue and the key characteristics and the tradeoffs of using the four main server extension techniques: SSI, SSS, CGI and API.

For now, lets take a look at the security issues involving APIs and their applications.

## Sockets

A socket is one end-point of a two-way communication link between two programs running on the network. For instance, a server application usually listens to a specific port waiting for connection requests from a client. When a connection request arrives, the client and the server establish a dedicated connection over which they can communicate. During the connection process, the client is assigned a local port number, and binds a socket to it. The client talks to the server by writing to the socket and gets information from the server by reading from it. Similarly, the server gets a new local port number, while listening for connection requests on the original port. The server also binds a socket to its local port and communicates with the client by reading from and writing to it. The client and the server must agree on a protocol before data starts being exchanged.

The following program is a simple example of how to establish a connection from a client program to a server program through the use of sockets, which was extracted from Sun's Web site at URL <http://java.sun.com/docs/books/tutorial/networking/sockets/readingWriting.html>. I encourage you check the site our for more in-depth information about it and the use of the API `java.net.Socket`, a very versatile API..

The `Socket` class in the `java.net` package is a platform-independent implementation of the client end of a two-way communication link between a client and a server. The `Socket` class sits on top of a platform-dependent implementation, hiding the details of any particular system from your Java program. By using the `java.net.Socket` class instead of relying on native



code, your Java programs can communicate over the network in a platform-independent fashion.

This client program, EchoTest, connects to the standard Echo server (on port 7) via a socket. The client both reads from and writes to the socket. EchoTest sends all text typed into its standard input to the Echo server by writing the text to the socket. The server echos all input it receives from the client back through the socket to the client. The client program reads and displays the data passed back to it from the server:

```
import java.io.*;

import java.net.*;

public class EchoTest {

    public static void main(String[] args) {

        Socket echoSocket = null;

        DataOutputStream os = null;

        DataInputStream is = null;

        DataInputStream stdIn = new DataInputStream(System.in);

        try {

            echoSocket = new Socket("taranis", 7);

            os = new DataOutputStream(echoSocket.getOutputStream());

            is = new DataInputStream(echoSocket.getInputStream());

        } catch (UnknownHostException e) {

            System.err.println("Don't know about host: taranis");

        } catch (IOException e) {

            System.err.println("Couldn't get I/O for the connection to: taranis");

        }

    }

}
```

```
if (echoSocket != null && os != null && is != null) {

try {

String userInput;

while ((userInput = stdin.readLine()) != null) {

os.writeBytes(userInput);

os.writeByte('\n');

System.out.println("echo: " + is.readLine());

}

os.close();

is.close();

echoSocket.close();

} catch (IOException e) {

System.err.println("I/O failed on the connection to: taranis");

}

}

}
```

Let's walk through the program and investigate the interesting bits.

The following three lines of code within the first try block of the main() method are critical--they establish the socket connection between the client and the server and open an input stream and an output stream on the socket:

```
echoSocket = new Socket("taranis", 7);
```

```
os = new DataOutputStream(echoSocket.getOutputStream());
```

```
is = new DataInputStream(echoSocket.getInputStream());
```

The first line in this sequence creates a new `Socket` object and names it `echoSocket`. The `Socket` constructor used here (there are three others) requires the name of the machine and the port number that you want to connect to. The example program uses the hostname `taranis`, which is the name of a (hypothetical) machine on our local network. When you type in and run this program on your machine, you should change this to the name of a machine on your network. Make sure that the name you use is the fully qualified IP name of the machine that you want to connect to. The second argument is the port number. Port number 7 is the port that the Echo server listens to.

The second line in the code snippet above opens an output stream on the socket, and the third line opens an input stream on the socket. `EchoTest` merely needs to write to the output stream and read from the input stream to communicate through the socket to the server. The rest of the program achieves this. If you are not yet familiar with input and output streams, you may wish to read `Input and Output Streams`.

The next section of code reads from `EchoTest`'s standard input stream (where the user can type data) a line at a time.

`EchoTest` immediately writes the input text followed by a newline character to the output stream connected to the socket.

```
String userInput;
```

```
while ((userInput = stdIn.readLine()) != null) {
```

```
os.writeBytes(userInput);
```

```
os.writeByte('\n');
```

```
System.out.println("echo: " + is.readLine());
```

```
}
```

The last line in the while loop reads a line of information from the input stream connected to the socket. The `readLine()` method blocks until the server echos the information back to `EchoTest`. When `readline()` returns, `EchoTest` prints the information to the standard output.

This loop continues--`EchoTest` reads input from the user, sends it to the Echo server, gets a response from the server and displays it--until the user types an end-of-input character.

When the user types an end-of-input character, the while loop terminates and the program continues, executing the next three lines of code:

```
os.close();
```

```
is.close();
```

```
echoSocket.close();
```

These lines of code fall into the category of housekeeping. A well-behaved program always cleans up after itself, and this program is well-behaved. These three lines of code close the input and output streams connected to the socket, and close the socket connection to the server. The order here is important--you should close any streams connected to a socket before you close the socket itself.

This client program is straightforward and simple because the Echo server implements a simple protocol. The client sends text to the server, the server echos it back. When your client programs are talking to a more complicated server such as an http server, your client program will also be more complicated. However, the basics are much the same as they are in this program:

1. Open a socket.
2. Open an input stream and output stream to the socket.
3. Read from and write to the stream according to the server's protocol.
4. Close streams.
5. Close sockets.

Only step 3 differs from client to client, depending on the server. The other steps remain largely the same.

But knowing how a socket works, even if using reliable codes such as the above, does not necessarily makes your system immune to security holes and threats. It all will depend on the environment you're in. Security holes generated by sockets will vary depending on what kind of threat they can allow, such as all:

- Denial of service
- The increase of privileges to local users without authorizations
- Access of remote hosts without authorization, etc.

## ***BSD sockets***

Daniel L. McDonald (Sun Microsystems, USA), Bao G. Phan (Naval Research Laboratory, USA) and Randall J. Atkinson (Cisco Systems, USA) wrote a paper entitled "A Socket-Based Key Management API (and Surrounding Infrastructure)," which can be found at the URL [http://info.isoc.org/isoc/whatis/conferences/inet/96/proceedings/d7/d7\\_2.htm](http://info.isoc.org/isoc/whatis/conferences/inet/96/proceedings/d7/d7_2.htm), that addresses the security concerns expressed by the Internet Engineering Task Force (IETF) in this area.

The IETF has advanced to Proposed Standard, a security architecture for the Internet Protocol [Atk95a, Atk95b, Atk95c]. The presence of these security mechanisms in the Internet Protocol does not, by itself, ensure good security. The establishment and maintenance of cryptographic keys and related security information, also known as key management, is also crucial to effective security. Key management for the Internet Protocol is a subject of much experimentation and debate [MS95] [AMP96a] [AMP96b] [Orm96]. Furthermore, key management strategies have a history of subtle flaws that are not discovered until after they are published or deployed [NS87].

McDonald, Phan and Atkinson paper proposes an environment which allows implementations of key management strategies to exist outside the operating system kernel, where they can be implemented, debugged, and updated in a safe environment. The Internet Protocol suite has gained popularity largely because of its availability in the Berkeley Software Distribution (BSD) versions of the Unix operating system. Even though many commercial operating systems no longer use the BSD networking implementation, they still support BSD abstractions for application programmers, such as the sockets API [LMKQ89]. The sockets interface allows applications in BSD to communicate with other applications, or sometimes, even with the operating system itself. One of the recent developments in BSD was the routing socket [SkI91], which allows a privileged application to alter a node's network routing tables.

This abstraction allows a BSD system to use an appropriate routing protocol, without requiring changes inside the kernel. Instead, routing protocols are implemented in user-space daemons, such as routed or gated.

## ***Windows sockets***

Windows Sockets Version 2.0, provides a powerful and flexible API for creating universal TCP/IP applications. You can create any type of client or server TCP/IP application with an implementation of Windows Sockets specification. You can port Berkeley Sockets applications and take advantage of the message-based Microsoft Windows programming environment and

paradigm.

**Tip:**

To know more about sockets, check the book "Network Programming with Windows Sockets," by Pat Bonner, which is a great helper. She writes with a tech-talk-avoiding clarity I've not seen in any other books on the subject.

WinSock 2 specification has two distinct parts: the API for application developers, and the SPI for protocol stack and namespace service providers. The intermediate DLL layers are independent of both the application developers and service providers. These DLLs are provided and maintained by Microsoft and Intel. The Layered Service Providers would appear in this illustration one or more boxes on top of a transport service provider.

**Tip:**

For more information about Windows Socket, check the URL <http://www.sockets.com>. The information you will find there can help you with your Windows Sockets (WinSock) application development. There are lots of useful information there, including sample source code, detailed reference files, and web links. Most of this material comes out of the book "Windows Sockets Network Programming," which provides a detailed introduction, and complete reference to WinSock versions 1.1 and 2.0.

### *c. Java APIs*

Java Enterprise APIs support connectivity to enterprise databases and legacy applications. With these APIs, corporate developers are building distributed client/server applets and applications in Java that run on any OS or hardware platform in the enterprise. Java Enterprise currently encompasses four areas: JDBCTM, Java IDL, Java RMI and JNDITM. For more information about this APIs I recommend you to check the JavaLink site at URL <http://java.sun.com/products/api-overview/index.html>.

Now, Joseph Bank ([jbank@mit.edu](mailto:jbank@mit.edu)), from MIT wrote a paper discussing the Java security issues. The document is available at URL <http://www.swiss.ai.mit.edu/~jbank/javapaper/javapaper.html>

Bank discusses the potential problems raised by executable content, such as in Java. As he comments, the advantages of executable content come from the increase in power and flexibility provided by software programs. The increased power of Java applets (the Java term for executable content) is also the potential problem. When a user is surfing the Web, they should not have to worry that an applet may be deleting their files or sending their private information over the network surreptitiously.

The essence of the problem is that running programs on a computer typically gives that program access to certain resources on the host machine. In the case of executable content, the program that is running is untrusted.

If a Web browser that downloads and runs Java code is not careful to restrict the access that the untrusted program has, it can provide a malicious program with the same ability to do mischief as a hacker who had gained access to the host machine. Unfortunately, the solution is not as simple as completely restricting a downloaded programs access to resources. The reason that one gives programs access to resources in the first place is that in order to be useful a program needs to access certain resources. For example a text editor that cannot save files is useless. Thus, if one desires to have useful and secure executable content, access to resources needs to be carefully controlled.

As Bank concludes in his paper, "the security measures of Java provide the ability to tilt this balance whichever way is preferable. For a system where security is of paramount importance, using Java does not make sense; it is not worth the added security risk. For a system such as a home computer, many people are likely to find that the benefits of Java outweigh the risks. By this same token, a number of systems are not connected to the Internet because it is a security risk that outweighs the benefits of using the Internet. Anyone that is considering using Java needs to understand that it does increase the security risk, but that it does provide a fairly good "firewall."

**c. *Perl modules***

Briefly described, Perl is a Practical Extraction and Reporting Language (Perl). Perl for Win32 is a port of most of the functionality in Perl, with some extra Win32 API calls thrown in so that you can take advantage of native Windows functionality, and runs on Windows 95 and Windows NT 3.5 and later.

There is a module with this package, Perl for ISAPI, which is a ISAPI DLL that runs Perl scripts in process with Internet Information Server (IIS) and other ISAPI compliant web servers. This provides better performance, at the risk of some functionality.

The following is a sample code written in PerlScript, extracted from ActiveWare Internet Corp, site, at URL <http://www.activestate.com/PerlScript/showsource.asp?filename=hello.asp&URL=/PerlScript/hello.asp>. This sample coding gives one an example of how versatile and portable this script is.

HTML Source for: /PerlScript/hello.asp

```
<%@ LANGUAGE = PerlScript %>
```

```
<html>
```

```
<HEAD>
```

```
<!--
```

## 1996, Microsoft Corporation. All rights reserved.

Developed by ActiveWare Internet Corp., <http://www.ActiveWare.com>

```
-->
```

```
<TITLE> Create a MSWC.BrowserType Browser Capabilities component </TITLE>
```

```
</HEAD>
```

```
<BODY> <BODY BGCOLOR=#FFFFFF>
```

```
<!--
```

ActiveWare PerlScript sample

PerlScript: The coolest way to program custom web solutions.

-->

<!-- Masthead -->

<TABLE CELLPADDING=3 BORDER=0 CELLSPACING=0>

<TR VALIGN=TOP ><TD WIDTH=400>

<A NAME="TOP"><IMG SRC="PSBWlogo.gif" WIDTH=400 HEIGHT=48 ALT="ActiveWare PerlScript" BORDER=0></A><P>

</TD></TR></TABLE>

<%

for (\$i = 3; \$i < 8; \$i++) {

%><font size=

<%= \$i %> > "Hello World!" </font><BR> <%

} %>

<!-- ++++++

here is the standard showsource link -

Note that PerlScript must be the default language --> <hr>

<%

\$url = \$Request->ServerVariables('PATH\_INFO')->item;

\$\_ = \$Request->ServerVariables('PATH\_TRANSLATED')->item;

s/[\//\\](\w\*\\.asp\Z)//m;

\$params = 'filename='."\$1".'&URL='."\$url";

```
$params =~ s#([^\a-zA-Z0-9&_\.:/-\]\]{1})#uc '%' . unpack('H2', $1)#eg;
```

```
%>
```

```
<A HREF="index.htm"> Return </A>
```

```
<A HREF="showsource.asp?<%= $params%>">
```

```
<h4><i>view the source</i></h4></A>
```

```
</BODY>
```

```
</HTML>
```

There is a lot written about Perl out there, and it doesn't make sense to discuss too much about Perl on a firewall book. Nevertheless, I would like to comment a little bit about Perl for Win32, by ActiveWare Internet Corp (<http://www.activeware.com/>), as it closely interacts with ISAPI, playing a role on APIs security.

Perl for Win32 refers to a port of the Perl programming language to the Win32 platform. Please note that Perl for Win32 does not run on Windows 3.11 and Win32s.

You should be careful with these modules, as most of them are distributed AS-IS, without any guarantee to work. If a module doesn't work, chances are:

- Some of the functions are not provided by Perl for Win32
- Some of the UNIX tools being used are not available on Win32 platforms, or
- It makes assumptions about the way files are handled that aren't valid on Win32 platforms

Also, be careful with the Perl for ISAPI build 307, which doesn't work, due to a problem with POST. Activeware asks that you continue to use build 306. As soon as this bug is fixed it should be announced on the Perl-Win32-Announce Mailing List.

### *c. CGI Scripts*

Typically, CGI scripts are insecure, and Perl CGI Scripts are not exception to the rule, especially affecting Web-centric application, such as browsers.

Take for example the Netscape server. It does not use Windows NT's File Manager's associations between file extensions and applications. Consequently, even though you may have associated the extension .pl with the Perl interpreter, Perl scripts are not recognized as such when placed in the cgi-bin directory. In order to workaround this problem, an earlier Netscape technical note suggested that you place the perl.exe file into the cgi-bin directory and refer to your scripts as /cgi-bin/perl.exe?&my\_script.pl.

However, it was a bad, very bad idea! This technique allowed anyone on the Internet to execute an arbitrary set of Perl commands right onto your server by just invoking such scripts as /cgi-bin/perl.exe?&-e+unlink+%3C\*%3E, which once run will erase all files stored in your server's current directory! This was bad news! A more recent Netscape technical note suggested then to encapsulate your Perl scripts in a .bat file. However, because of a related problem with batch scripts, this is still not safer.

Because the EMWACS, Purveyor and WebSite NT servers all use the File Manager extension associations, you can execute perl scripts on these servers without placing perl.exe into cgi-bin. They are safe from this bug.

The NCSA httpd is also affected by CGI scripts with security holes. The NCSA httpd prior to version 1.4 contain a serious



security hole relating to a fixed-size string buffer, which allows for remote users to break into systems running this server by requesting an extremely long URL. Even though this is a bug already well publicized for more than couple year, many sites are still running unsafe versions of this server. From version 1.5 on the bug was fixed.

But not so long ago, it was found that the example C code (cgi\_src/util.c) usually distributed with the NCSA httpd as a boiler-plate for writing safe CGI scripts omitted the newline character from the list of characters. This omission introduced a serious bug into CGI scripts built on top of this template, which caused a security hole where a remote user could exploit this bug to force the CGI script to execute any arbitrary UNIX command. This is another example of the dangers of executing shell commands from CGI scripts.

The Apache server, versions 1.02 and earlier, also contains this hole in both its cgi\_src and src/ subdirectories. The patch to fix these holes in the two util.c files is not complicated. You will have to recompile the "phf" and any CGI scripts that use this library after applying the GNU patch, which can be found at URL <ftp://prep.ai.mit.edu/pub/gnu/patch-2.1.tar.gz>.

Here it is the source:

```
tulip% cd ~www/ncsa/cgi_src
```

```
tulip% patch -f < ../util.patch
```

```
tulip% cd ../src
```

```
tulip% patch -f < ../util.patch
```

```
----- cut here -----
```

```
*** ./util.c.old      Tue Nov 14 11:38:40 1995
```

```
--- ./util.c          Thu Feb 22 20:37:07 1996
```

```
*****
```

```
*** 139,145 ****
```

```
l=strlen(cmd);
```

```
for(x=0;cmd[x];x++) {
```

```
! if(ind("&`'\\"|*?~<>^()[]{}$\\",cmd[x]) != -1){
```

```
for(y=l+1;y>x;y--)
```

```
cmd[y] = cmd[y-1];
```

```
l++; /* length has been increased */
```

```
--- 139,145 ----
```

```
l=strlen(cmd);
```

```
for(x=0;cmd[x];x++) {
```

```
! if(ind("&`'\\"|*?~<>^()[]{}$\\n",cmd[x]) != -1){
```

```
for(y=l+1;y>x;y--)
```

```
cmd[y] = cmd[y-1];
```

```
l++; /* length has been increased */
```

```
----- cut here -----
```

### **(c) ActiveX**

As I mentioned on chapter 5, "Firewalling Challenges: The Advanced Web," You should not consider and ActiveX applet secure.

However, you should understand that ActiveX is only as secure as its architecture, design, implementation and environment permits. Although Microsoft never tried to state the security of ActiveX, as its use of digital signatures is intended only to the extent that it allows you to prove who was the originators of the applet. As a commented on chapter 5, if Microsoft was attempting to do any further security on ActiveX, the WinVerifyTrust API implementation would be checking the signature against the CA every time the object was accessed. But again, dealing with certificate revocation is a lot of work!

The way its implemented, this check is done once and recorded, subsequent access check first to see if its been previously authorized, and if so, it will use the object. So if a CA invalidates an object, anyone who had previously accessed the object would continue to use the malicious object without question. But you don't have to rely on it in order to grant some level of security. As it is discussed on chapter 14, "Types of Firewalls," there are products nowadays filtering ActiveX and Java applets.

But don't put all your eggs into a single nest, or firewall. Of course, firewalls are needed, but you will also need virus scanning, applet filters, encryption and so on. Also, you must understand that all these security technologies are simply an artifact of our inability (lack of time, knowledge, money, who knows?) to did deeper into the foundation of any security model: a complete and well elaborated security policy, which is followed and enforced. It's usually because we don't want to deal with it that we look for fixes such as firewalls, etc. Thus, you must understand that these products and techniques are tools, and you'll need to come up with the "intel," the knowledge anyway.

A firewall should be for you what a word processor is for me when I write this book. It doesn't matter if I use a Pentium Pro or a 486 PC, with a word processor such as Microsoft Word or FrameMaker to write the book. Surely these tools will help me to spell the text, write faster and so on, but if I don't have a clear picture of what I want to accomplish with my writings, nothing will help me.

### **d. ActiveX DocObjects**

The new ActiveX DocObjects technology allows you to edit a Word document on Internet Explorer (IE), by just selecting a

hyperlink displayed by Internet Explorer (IE). After clicking on a hyperlink to a Microsoft Word document, the document is displayed in Internet Explorer's window. The Word menus and toolbars are displayed along with those from Internet Explorer, as shown on Figure 06.1, extracted from MacMillan/Sams.Net site at URL <http://www.mcp.com/sams/books/156-4/pax11.htm#I21>.

This is what Microsoft calls visual editing, where Microsoft Word becomes activated in Internet Explorer's window. The editing functions of both applications coexist on the Internet completely intact.

You can benefit greatly from this type of "online" document management and editing. It gives you a much greater maintenance capability, which doesn't exist for most file formats. The problem of distributing your documentation is also alleviated merely by putting the documents on the Web.

This technology is intended for use in both Internet Explorer 3.0 and in the Office95 Binder for visually editing various file formats. The ActiveX DocObject technology uses a modified menu-sharing technique that more closely resembles the data's own standalone native editing application.

What you should be careful here is that by clicking on a hyperlink to open a Word document on the Web could trigger a malicious applet. The same is true for Adobe's PFD files. When you click on a link or filename on the Web, you don't know what this document contains, or even if it will open a Word document.

### ***c. Distributed Processing***

Distributed Processing (DP) is the process of distribution of applications and business logic across multiple processing platforms, which implies that processing will occur on more than one processor in order for a transaction to be completed. Thus, the processing is distributed across two or more machines and the processes are most likely not running at the same time, as each process performs part of an application in a sequence.

Often the data used in a distributed processing environment is also distributed across platforms.

Don't confuse distribute processing with cooperative processing, which is the computing which requires two or more distinct processors to complete a single transaction. Cooperative processing is related to both distributed and client/server processing. It is a form of distributed computing where two or more distinct processes are required to complete a single business transaction. Usually, these programs interact and execute concurrently on different processors.

Cooperative processing can also be considered to be a style of client/server processing if communication between processors is performed through a message passing architecture.

Lets take a look at some examples of it.

### ***d. XDR/RPC***

XDR/RPC are routines used for describing the RPC messages in XDR language. They should normally be used by those who do not want to use the RPC package directly. These routines return TRUE if they succeed, FALSE otherwise.

XDR routines allow C programmers to describe arbitrary data structures in a machine-independent fashion. Data for remote procedure calls (RPC) are transmitted using these routines.

#### **Tip:**

For a list of the XDR routines check the URL <http://www.doc.ic.ac.uk/~mac/manuals/solaris-manual-pages/solaris/usr/man/man3n/xdr.3n.html>, from which the above definition was extracted.

### ***d. RPC***

The rpc file is a local source containing user readable names that can be used in place of RPC program numbers. The rpc file can be used in conjunction with or instead of other rpc sources, including the NIS maps "rpc.byname" and "rpc.bynumber" and the NIS+ table "rpc".

The rpc file has one line for each RPC program name. The line has the following format:

name-of-the-RPC-program RPC-program-number aliases

Items are separated by any number of blanks and/or tab characters. A "\_" indicates the beginning of a comment; characters up to the end of the line are not interpreted by routines which search the file.

RPC-based middleware is a more general-purpose solution to client/server computing than database middleware. Remote procedure calls are used to access a wide variety of data resources for use in a single application.

Messaging middleware takes the RPC philosophy one step further by addressing the problem of failure in the client/server system. It provides synchronous or asynchronous connectivity between client and server, so that messages can be either delivered instantly or stored and forwarded as needed.

Object middleware delivers the benefits of object-oriented technology to distributed computing in the form of object request brokers. ORBs package and manage distributed objects, which can contain much more complex information about a distributed request than an RPC or most messages and can be used specifically for unstructured or nonrelational data.

#### ***d. COM/DCOM***

Database middleware as mentioned above, is used on database-specific environments. It provides the link between client and server when the client application that accesses data in the server's database is designed to use only one database type.

TP monitors have evolved into a middleware technology that can provide a single API for writing distributed applications. Transaction-processing monitors generally come with a robust set of management tools that add mainframe-like controls to open distributed environments.

Proprietary middleware is a part of many client/server development tools and large client/server applications. It generally runs well with the specific tool or application environment it is a part of, but it doesn't generally adapt well to existing client/server environments, tools, and other applications.

There are other technologies such as CORBA, ILU that also supports database middleware. For more information check the middleware glossary of LANtimes at <http://www.lantimes.com/lantimes/95aug/508b068a.html>.

[Orders](#)
[Orders](#)
[Back](#)
[Forward](#)
[Comments](#)
[Comments](#)


---

[COMPUTING MCGRAW-HILL](#) | [Beta Books](#) | [Contact Us](#) | [Order Information](#) | [Online Catalog](#)

---

HTML conversions by [Mega Space](#).

This page updated on December 05, 1997 by [Webmaster](#).

[Computing McGraw-Hill](#) is an imprint of the [McGraw-Hill Professional Book Group](#).

Copyright ©1997 [The McGraw-Hill Companies, Inc.](#) All Rights Reserved.

Any use is subject to the rules stated in the [Terms of Use](#).

|        |        |
|--------|--------|
| Orders | Orders |
|--------|--------|

|      |         |
|------|---------|
| Back | Forward |
|------|---------|

|          |          |
|----------|----------|
| Comments | Comments |
|----------|----------|

© 1997 The McGraw-Hill Companies, Inc. All rights reserved.  
Any use of this Beta Book is subject to the rules stated in the [Terms of Use](#).

---

## Chapter 7

# ***What is an Internet/Intranet Firewall After All?***

As I wrote in a book I co-authored with Ablan and Yanoff, by Sams.Net (*Web Site Administrator's Survival Guide*), the first time I heard about firewalls was with my mechanic. Seriously! He was explaining to me that cars have this part that separates the engine block from the passenger compartment, and it's called a *firewall*. If the car explodes, the firewall protects the passengers.

Similarly, a firewall in computer terms protects your network from untrusted networks. On one side you have a public network, without any kind of control over what is being done, how or where. On the other side you have the production network of a company with a corporate network that must be protected against any damaging action. Some even question: if we really need to protect a corporate network, so then why allow a network of public domain, such as the Internet, to access it.

The reason is simple: It's a matter of survival! Companies rely more on the Internet to advertise their products and services. The Internet is growing tremendously, just like big market places and shopping malls, more people are coming to the Internet. The more that come, the more security is necessary to guarantee the integrity of products sold, as well as the safety of those participating in this market (a.k.a., electronic commerce). It has become necessary to protect data, transmissions and transactions from any incidents, regardless if the cause is unintentional or by malicious acts.

This chapter discusses the mechanisms used to protect your corporate network/Intranet and/or Web servers against unauthorized access coming from the Internet or even from inside a protected network. It also reviews what firewalls are after all and how important they are in providing a safe Internet connection. You will learn about the following:

- The purpose of firewalls
- Advantages and disadvantages of firewalls
- Basic design decisions
- Threats and countermeasures provided by firewalls
- Major firewall vendors (a more detailed list is in the Appendix)

- Firewall procurement and administration
- How to install a typical firewall

# What are Firewalls After All?

There are several models and configurations of firewalls out there, but the basic idea behind it is always the same. You need to allow users from a protected network, such as your corporate network, to access a public network, such as the Internet, and at the same time, to make available to this public network the services and products of the company.

The problem is that when your company is connected to the Internet without proper security measurements in place, you become exposed to attacks from other servers on the Internet. Not only does your corporate network become vulnerable to unauthorized access, but also do all other servers in your corporate network.

Therefore, when you begin to plan out how you will protect your network from the many threats the Internet can bring you will start by thinking about firewalls. However, even before thinking about it, you must define how and which services and information you will make available to the Internet. Evidently, and first of all, you will want to make sure that your server is secure. You will have to be able to block unauthorized login access, file transfer access, and remote command execution, and perhaps even deny services such as Rlogin, telnet, (t)FTP, SMTP, NFS, and other RPC services. Once you start to use, or have access to these services, that's when you will need to build a firewall. Figure 7.1 gives you a basic idea of a firewall and its purpose.

But what is a firewall anyway? Basically, a firewall separates a protected network from an unprotected one, the Internet. It screens and filters all connections coming from the Internet to the protected (corporate) network, and vice versa, through a single, concentrated security checkpoint. A firewall makes sure that you cannot reach the Internet from the internal network, nor vice versa, unless you pass through this checkpoint (also known as choke-hold firewall). Some systems even require you to telnet the firewall.

But even before you define what type of firewall best suites your needs you will need to analyze the topology of your network to determine if the various components of your network, such as hubs, switches, routers and cabling are suitable for an specific firewall model. Chapter 10, "Putting It Together: Firewall Design and Implementation," provides information on what to look for in a firewall, according to your corporate environment and security needs. Also, chapter 14, "Types of Firewalls," provides an extensive list of the main firewall products and in-depth information about their technology, security and management features.

To better understand a firewall and its purpose you should have a more detailed understanding of what a firewall and this protection barrier is. You need look at your corporate network based on the layers of its International Standards Organization (ISO) model. There you find the repeaters and hubs acting at the first layer, switches and bridges acting at the second layer and routers at the third layer. A firewall passes through all these layers as it acts at the sixth and seventh level, the layers responsible for the session establishment controls and applications. Thus, with a firewall we can control the flow of information throughout the establishment of sessions or even by determining which operations will or not be allowed.

However, as you will see, a firewall does more than protect you against the electronic version of airbrushing someone else's wall or breaking glass windows on the digital street. It will help you manage a variety of aspects on your *gate* to the Web by keeping the jerks out while enabling you to concentrate on your job.

## The Purpose of a Firewall

Say you have just acquired a car. It's blue with four doors. Is an alarm enough to secure it? In case the car disappears, its color and the fact it has four doors won't make much difference. I'm sure you wouldn't be so casual about it. You probably would have insurance for it and would list its vehicle identification number, any accessories it has, plate numbers, and so on. But believe it or not, many companies treat the security of their network assets especially data communication and internetworking assets very lightly. Often there will be no policies, or any sort of record keeping, where the security of their systems are treated with much less information than you would with your car.

That's where firewalls come in, but you must realize that a firewall alone will not secure your network. It is only part of a broader area in protecting your web site and networking in general.

In order to secure your corporate network, you must define your idea of a network perimeter. You need to determine what things must be protected, develop a security policy, and establish mechanisms to enforce the policy and methods you are going to employ. Of course, there are mechanisms besides the firewall that you can add to tremendously increase your level of security.

These mechanisms must come after your security policy is developed, not before. This should be the main idea you should retain for this section: to define a security mechanism that will protect your corporate site, in specific firewalls, and to provide you with the prerequisites to implement it. Policies and procedures are one indispensable prerequisite. The methods you are going to employ and your analysis of the results are another.

Many companies and data centers are guided by computing security policies, particularly those organizations in the public sector that are likely to be a target, such as the Department of Defense (DoD) and other government agencies. Procedures are established that must be adhered to. Curiously, it rarely happens with the private sector, especially when it comes to connecting to the Internet. You would be surprised to learn that many private companies very often neglect the development of a security policy, and therefore their security mechanisms are weak if not faulty.

Security policies vary from organization to organization, of course, but one issue that will set these policies aside will be the platform for what they are being developed. A firewall can be implemented on UNIX, NT, DOS or a proprietary platform. You must look closely at the platform you will be choosing, as it will definitely define all future projects, level of security and consequently the security policy being developed. That's why a security policy must come first to guarantee the success of the mechanisms that will be implemented.

As a LAN or Web administrator, you already know that the hardest part of connecting your corporation to the Internet is not justifying the expense or effort, but convincing management that it is safe to do so, especially at large companies. A firewall not only adds real security, but also plays an important role as a security blanket for management.

Furthermore, have you ever thought about the functions of a United States Embassy in other countries? A firewall can act just like one. As your corporate *ambassador* to the Internet, a firewall can control and document the *foreign affairs* of your organization.

**Note:**

If you want more information on firewalls, a great site to visit is URL <ftp://ftp.greatcircle.com/ub/firewalls>. A firewall toolkit and papers are available at <ftp://ftp.tis.com/ub/firewalls>.

## ***The Firewall Role of Protection***

A firewall greatly improves network security and reduces risks to servers on your network by filtering inherently insecure services. As a result, your network environment is exposed to fewer risks because only selected protocols are able to pass through the firewall.

For example, a firewall could prohibit certain vulnerable services such as NFS from entering or leaving a protected network. This provides the benefit of preventing the services from being exploited by outside attackers, but at the same time permits the use of these services with greatly reduced risk of exploitation. Services such as NIS or NFS that are particularly useful on a Local Area Network basis can thus be enjoyed and used to reduce the server management burden.

Firewalls can also provide protection from routing-based attacks, such as source routing and attempts to redirect routing paths to compromised sites through ICMP redirects. It could reject all source-routed packets and ICMP redirects and then inform administrators of the incidents.

The problem with firewalls, though, is that they limit access to and from the Internet. In some configurations, you may decide to use a proxy server, which is explored in more detail later to filter the inbound and outbound access your policy has determined to be safe

## ***Firewalls Providing Access Control***

A firewall can provide access control to site systems. For instance, some servers can be made reachable from outside networks, whereas others can be effectively sealed off from unwanted access. Depending on the level of risk you are willing to take in your corporate network, watch for outside access to the internal network servers, except for special cases such as mail servers or RAS services.

Also, watch for viruses being downloaded by users via e-mail and spread throughout the protected network. Another threat are that so called applets (mainly Java, JavaScript and ActiveX), that could initiate remote process on a workstation that could affect a server or even disable a firewall.



**Tip:**

Read more about ActiveX and Java vulnerabilities and security holes on chapter 5, "Firewalling Challenges: The Advanced Web," under the section "Code in the Web."

This brings us to one of the main purposes an access policy is particularly adept at enforcing: Never provide access to servers or services that do not require access as they could be exploited by hackers since the access is not necessary or required.

## The Security Role of a Firewall

A firewall can actually be less expensive for an organization in that all (or most) modified software and additional security software can be located on the firewall system than if distributed on each server or machines. In particular, one-time-password systems and other add-on authentication software can be located at the firewall rather than be on each system that needs to be accessed from the Internet.

Also, don't neglect internal security. Very often too much emphasis is giving to the firewall, but if a hacker cracks in, unless you have some internal security policy in place, your network will be exposed. That's why many times it might be a good idea to place the server to be available to the Internet outside of the firewall. The services on this server will very likely become exposed to the Internet threats, but you could easily have a replica of the server inside the firewall and available for quick recovery. You could also keep all the systems configuration of the external server in a CD-ROM, making it secure against modifications.

Other solutions to your corporate network security could involve modifications at each server system. Although many techniques are worthy of consideration for their advantages and are probably more appropriate than firewalls in certain situations, firewalls tend to be simpler to implement, because only the firewall needs to run specialized software, unless if you have a package filtering firewall or require your users to Telnet it. In this case, you either will need a router filtering the packages or a dedicated machine.

## Promoting Privacy with a Firewall

Privacy should be of great concern for every corporate network because what normally would be considered innocuous information might actually contain clues that would be useful to a hacker. By using a firewall, your site can block access from services such as Finger and Domain Name Service, for example. Finger, to refresh, displays information about users such as their last login time, whether they've read mail, and other items. But Finger can also reveal information to hackers about how often a system is used, whether the system has active users connected, and whether the system could be attacked without attracting the attention of administrators and other monitoring systems.

Another advantage of using a firewall at your site is that by having all access to and from the Internet passing through a firewall, you can log accesses and provide valuable statistics about network usage.

# Advantages and Disadvantages of Firewalls

Besides logins and statistics, there are many other advantages to using firewalls. Despite these advantages, you should be aware that there are also a number of disadvantages: things that firewalls cannot protect against, such as access restrictions, back-doors treats (modem and/or RAS servers by-passing the firewall), and vulnerability to inside hackers, to name a few. Chapter 10, "Putting it Together: Firewall Design and Implementation" provides a summary of the pros and cons of the different types of firewalls available.

## ***Access Restrictions***

Obviously, a firewall will very likely block certain services that users want, such as Telnet, FTP, X Window, NFS, and so on. These disadvantages are not unique to firewalls alone, however network access could be restricted at the server level as well, depending on a site's security policy. A well-planned security policy that balances security requirements with user needs can help greatly to alleviate problems with reduced access to services.

Nonetheless, some sites might lend themselves to a firewall due to their topology, or maybe due to services such as NFS, which could require a major restructuring of network use. For instance, you might depend on using NFS and NIS across major gateways. In this case, your relative costs of adding a firewall would need to be compared against the cost of exposure from not using a firewall.

## ***Back-Door Challenges: The Modem Threat***

By now, you have figured that existing back doors in your corporate network are not protected by firewalls. Therefore, if you have any unrestricted modem access, it is still an open door for hackers who could effectively use the access to bypass the firewall. Modems are now fast enough to make running *SLIP* (Serial Line IP) and *PPP* (Point-to-Point Protocol) feasible. A SLIP or PPP connection inside a protected subnet can also very easily become a potential back door. So if you are going to allow SLIP or PPP to exist without any kind of monitoring, why bother to have a firewall?

## ***Risk of Insider Attacks***

Generally, there is not much protection a firewall can provide against inside threats. Although a firewall might prevent outsiders from obtaining sensitive data, it does not prevent an insider from copying files or stealing information.

It is not safe to assume that a firewall provides protection from insider attacks. It would be unwise for you to invest resources in a firewall if you don't close the door of your systems to insider attacks as well.

Despite these disadvantages, I strongly recommend that you protect your site with firewalls.

# Firewall Components

The basic components in building a firewall include:

- Policy
- Advanced authentication
- Packet filtering
- Application gateways

The following topics give you a brief overview of each of these components and how they affect your site's security and, consequently, the implementation of your firewall.

## ***Network Security Policy***

The decision to set up a firewall can be directly influenced by two levels of network policy:

- Installation
- Use of the system

The network-access policy that defines services that will be allowed or explicitly denied from the restricted network is the high-level policy. It also defines how these services will be used. The lower-level policy defines how the firewall will actually restrict access and filter the services defined in the higher-level policy. However, your policy must not become an isolated document sitting in a drawer or a shelf it would be useless. The policy needs to become part of your company's security policy. Let's take a brief look at different types of security policies.

### **Flexibility Policy**

If you are going to develop a policy to deal with Internet access, web administration, and electronic services in general, it must be flexible. Your policy must be flexible because:

- The Internet itself changes every day at a rate that no one can follow including books, by the way. As the Internet changes, services offered through the Internet also change. With that, a company's needs will change also, so you should be prepared to edit and adapt your policy accordingly without compromising security and consistency. But remember: a security policy almost never change, but procedures should always be reviewed!
- The risks your company faces on the Internet are not static, either. They change every moment, always growing. You should be able to anticipate these risks and adjust the security processes accordingly.

### **Service-Access Policy**

When writing a service-access policy, you should concentrate on your companies' user issues as well as dial-in policy, SLIP connections, and PPP connections. The policy should be an extension of your organizational policy regarding the protection of Information Systems (IS) resources in your company. Your service-access policy should be realistically complete. Make sure you have one drafted before

implementing a firewall. The policy should provide a balance between protecting your network and providing user access to network resources.

## Firewall Design Policy

A *firewall design policy* is specific to the firewall. It defines the service-access policy implementation rules. You cannot design this policy without understanding the firewall capabilities and limitations, as well as the threats and vulnerabilities associated with TCP/IP. As mentioned earlier, firewalls usually do one of the following:

- Permit any service unless it is expressly denied.
- Deny any service unless it is expressly permitted.

A firewall that implements the first policy allows all services to pass into your site by default, except for those services that the service-access policy has determined should be disallowed. By the same token, if you decide to implement the second policy, your firewall will deny all services by default but then will permit those services that have been determined as allowed.

As you will surely agree, to have a policy that permits access to any service is not advisable because it exposes the site to more threats.

Notice the close relationship between the high-level service-access policy and the lower-level one. This relationship is necessary because the implementation of the service-access policy depends on the capabilities and limitations of the firewall systems you are installing as well as the inherent security problems that your Web services bring.

For example, some of the services you defined in your service-access policy might need to be restricted. The security problems they can present cannot be efficiently controlled by your lower-level policy. If your company relies on these services, which usually Web sites do, you probably will have to accept higher risks by allowing access to those services. This relationship between both service-access policies enables their interaction in defining both the higher-level and the lower-level policies in a consistent and efficient way.

The service-access policy is the most important component in setting up a firewall. The other three components are necessary to implement and enforce your policy. Remember: the efficiency of your firewall in protecting your site will depend on the type of firewall implementation you will use, as well as the use of proper procedures and the service-access policy.

## Information Policy

As an Internet manager, or even LAN or web administrator, if you intend to provide information access to the public, you must develop a policy to determine the access to the server (probably a web server) and include it in your firewall design. Your server will already create security concerns on its own, but it should not compromise the security of other protected sites that access your server.

You should be able to differentiate between an external user who accesses the server in search for information and a user who will utilize the e-mail feature, if you are incorporating one, for example, to communicate with users on the other side of the firewall. You should treat these two types of traffic

differently and keep the server isolated from other sites in the system.

## Dial-in and Dial-out Policy

Remote-access systems add useful features to authenticated users when they are not on-site or cannot access certain services or information through the company's web site. However, users must be aware of the threat of unauthorized access that a dial-in capability can generate.

You must be able to demonstrate the vulnerabilities that this feature will create if users are not cautious when accessing the internal network through a modem. A user's dial-out capability might become an intruder dial-in threat.

Therefore, you must consider dial-in and dial-out capabilities in your policy when designing your firewall. You must force outside users to pass through the advanced authentication of the firewall. This should be stressed in your policy, as well as the prohibition against unauthorized modems attached to any host or client that were not approved by MIS (Management of Information Systems) or are not passing through the firewall. Your goal is to develop a policy strong enough to limit the number of unauthorized modems throughout the company. By combining such a policy with an efficient pool of modems, you will be able to reduce the danger of hacker attacks on your company using modems as well as limiting your vulnerability.

Another factor you should consider involves web servers. Worse than having a modem line that enables dial-in and dial-out capabilities is the use of *serial line IP* (SLIP) or *Point-to-Point Protocol* (PPP) through the Web server or any other means of access to the company network. By far, it is a more dangerous back door to your system than modems could ever be, unless, of course, you pass it through the firewall.

## Advanced Authentication

Despite all of the time and effort writing up policies and implementing firewalls, many incidents result from the use of weak or unchanged passwords.

Passwords on the Internet can be *cracked* in many ways. The best password mechanism will also be worthless if you have users thinking that their login name backwards or a series of Xs are good passwords!

The problem with passwords is that once an algorithm for creating them is specified, it merely becomes a matter of analyzing the algorithm in order to find every password on the system. Unless the algorithm is very subtle, a cracker can try out every possible combination of the password generator on every user on the network. Also, a cracker can analyze the output of the password program and determine the algorithm being used. Then, he just need to apply the algorithm to other users so that their passwords can be determined.

Furthermore, there are programs freely available on the Internet to crack user's passwords. *Crack*, for example, is a program written with the sole purpose of cracking insecure passwords. It is probably the most efficient and friendly password cracker available at no cost. It even includes the ability to let the user specify how to form the words to use as guesses at user's passwords. Also, it has a built-in networking capability, which allows the load of cracking to be spread over as many machines as are

available on the network.

Also, you should be aware that some TCP or UDP services authenticate only to the level of server addresses and not to specific users. An NFS server, for example, cannot authenticate a specific user on a server it must grant access to the entire server. As an administrator, you might trust a specific user on a server and want to grant access to that user, but the problem is that you have no control over other users on that server, and will be forced to grant access to all users. It's all or nothing!

The risk you take is that a hacker could change the server's IP address to match that of the trusted client (the user you trust). The hacker could then construct a source route to the server specifying the direct path that IP packets should take to your web server and from the server back to the hacker's server all this using the trusted client as the last hop in the route to your server. The hacker sends a client request to the web server using the source route. Your server accepts the client request as if it came directly from the trusted client, and returns a reply to the trusted client. The trusted client, using the source route, forwards the packet on to the hacker's server. This process is called *IP spoofing*.

Figure 7.2 shows a basic example of a spoofed source IP address attack. Even though most routers can block source-routed packets, it's still possible to route packets through filtering-router firewalls if they are not configured to filter incoming packets whose source address is in the local domain. This attack is possible even if no reply packets can reach the attacker.

The following are examples of configurations that are potentially vulnerable to those attacks:

- Routers to external networks supporting multiple internal interfaces;
- Routers with two interfaces supporting subnets on the internal network, and
- Proxy firewalls where the proxy applications use the source IP address for authentication.

Please note that in the figure 7.2 the attack shown won't work if you have a properly configured router. Before about couple year ago, it would have worked, but after Kevin Mitnick, all the router vendors came out with fixes and told their customers to implement them. Most have, but the illustration is still valid though as many UNIX servers will still accept source-routed packets and pass them on as the source route indicates. Routers will accept source-routed packets as well, although most routers can block source-routed packets.

Couple years ago, the Internet Computer Emergency Response Team (CERT) sent out a security alert describing how hackers were using IP spoofing to break into many Internet sites. More than 23 million university, businesses, government facilities, and home computers connected to the Internet are exposed to the threat of having information stolen, systems *time-bombed*, and data corrupted through worms, Trojan horses, and viruses. All this, most of the time, for fun.

These kinds of attacks are usually aimed at applications that use authentication on source IP addresses. When the hacker can *pass* the packet, access to unauthorized data will be totally available. Keep in mind that the hacker doesn't have to get a reply packet back this break-in is possible even without it. Moreover, some network administrators tend to believe that disabling source routing at the router would prevent it. Not so! It cannot protect the internal network from itself.

If you have a router to external networks that supports multiple internal interfaces, you should consider a firewall because you are potentially exposed to hacker spoofing attacks. The same is true for routers with two interfaces supporting subnets on the internal network, as well as proxy firewalls if the proxy

applications use the source IP address for authentication.

Usually, what the hackers want is to access the root directory of your UNIX box. Once inside, they can dynamically replace telnetd and/or login, which enable them to capture existing terminal and login connections from any user on the system. This enables them to bypass the authentication schemes.

According to CERT, there are two steps you can take in order to prevent this kind of attack:

- Install a filtering router that will restrain the input to the external interface if it identifies the packet source as coming from inside the network. Even if it's an authenticated one, it won't go through.
- Filter outgoing packets to determine if the address is different from the internal network, so that attacks originated from inside can be prevented.

Figure 7.3 describes the CERT's recommendation for preventing IP spoofing. In this model, any external incoming packets must go through an additional router installed between the external interface (A) and the outside connection (B). This new intermediary router should be configured to block packets that have internal source address (C) on the outgoing interface connected to the original router.

If lack of security is risky, excessive complexity in configuration and controls is also. Use common sense, use the KISS (keep it simple... err steward) method. Server-access controls can be complex to configure and test.

One of the first things you should probably tell your Internet users is to choose passwords that are difficult to guess. You also tell them not to share their passwords with anyone. However, most users don't follow this advice, and even if they did, hackers can monitor passwords that are transmitted. One of the most effective alternatives to fight the hacker is to adopt advanced authentication measures.

Smartcards, such as credit card-like ID cards and other magnetic encoded cards, and software-based mechanisms are alternatives to cope with the weaknesses of traditional passwords. If you adopt one of these advanced authentication devices, hackers will not be able to reuse a password that was monitored during a connection. If you consider all of the inherent problems with passwords on the Internet, an Internet-accessible firewall that does not include some kind of advanced authentication system does not make much sense. The few mistakes and threats discussed previously give you an idea of what you are facing when announcing your new web site.

Some of the more popular advanced authentication devices in use today are called *one-time password systems*. A *smartcard*, for example, generates a response as an authenticator instead of a traditional password. It works in conjunction with software or hardware. Even if monitored, it can be used only once. The firewall's advanced authentication system should be located in the firewall because it centralizes and controls access to the site. You could install it on another server, but loading it on a firewall makes it more practical and manageable to centralize the measures.

Figure 7.4 illustrates what happens when advanced authentication is present. All connections and requests for sessions such as Telnet or FTP originating from the Internet to site systems must pass the advanced authentication before permission is granted. Passwords might still be required, but before permitting access, these passwords would be protected even if they were monitored.

# Packet Filtering

Usually, IP packet filtering is done using a router set up for filtering packets as they pass between the router's interfaces. These routers can filter IP packets based on the following fields:

- Source IP address
- Destination IP address
- TCP/UDP source port
- TCP/UDP destination port

Although not all packet-filtering routers can filter the source TCP/UDP port, most of them already have this capability. Some routers examine which of the routers network interfaces a packet arrived at, and this is used as an extra criterion. Unfortunately, most UNIX servers do not provide packet-filtering capability.

In order to block connections from or to specific web servers or networks, filtering can be applied in many ways, including the blocking of connections to specific ports. For instance, you might decide to block connections from addresses or sites that you consider being untrustworthy, or you might decide to block connections from all addresses external to your site all this can be accomplished by filtering. You can add a lot of flexibility simply by adding TCP or UDP port filtering to IP address filtering.

Servers such as the Telnet daemon usually reside at specific ports. If you enable your firewall to block TCP or UDP connections to or from specific ports, you will be able to implement policies to target certain types of connections made to specific servers but not others.

You could, for example, block all incoming connections to your web servers except for those connected to a firewall. At those systems, you might want to allow only specific services, such as SMTP for one system and Telnet or FTP connections to another system. Filtering on TCP or UDP ports can help you implement a policy through a packet-filtering router or even by a server with packet-filtering capability. Figure 7.5 illustrates packet-filtering routers on such services.

You can set up a ruleset to help you outline the permissions. Figure 7.6 shows a very basic, example ruleset of packet filtering. Actual rules permit more complex filtering and greater flexibility.

The first rule allows TCP packets from any source address and port greater than 1023 on the Internet to enter the destination address of 123.4.5.6 and port of 23 at the site. Port 23 is the port associated with the Telnet server, and all Telnet clients should have unprivileged source ports of 1024 or higher.

The second and third rules work in a similar way, except that packets to destination addresses 123.4.5.7 and 123.4.5.8, and port 25 for SMTP, are permitted.

The fourth rule permits packets to the site's NNTP server, but only from source address 129.6.48.254 to destination address 123.4.5.9 and port 119 (129.6.48.254 is the only NNTP server that the site should receive news from; therefore, access to the site for NNTP is restricted to that system only).

The fifth rule permits NTP traffic, which uses UDP as opposed to TCP, from any source to any destination address at the site.

Finally, the sixth rule denies all other packets. If this rule wasn't present, the router might not deny all subsequent packets.



Although packet filtering can effectively block connections from or to specific hosts, which increases your level of security substantially, packet-filtering routers have a number of weaknesses. Their rules are complex to specify and tough to test, because you either have to employ exhaustive testing by hand or find a facility where you can test the correctness of their rules. Logging capability is not found in all routers. If the router doesn't have this capability, you won't know if dangerous packets are passing through until it is too late.

Besides, in order to allow certain types of access (that normally would be blocked) to go through, you might have to create an exception to your rules. Exceptions sometimes can make filtering rules very difficult, or even unmanageable. How? Lets suppose you specify a rule to block all inbound connections to port 23 (the Telnet server). Assuming that you made exceptions such as accepting Telnet connections directly, a rule for each system needs to be added, right? Well, sometimes this kind of addition can complicate the entire filtering scheme! Don't forget: Testing complex sets of rules for correctness might be so difficult that you could never be able to set it right.

Another inconvenience to watch for is that some packet-filtering routers will not filter on the TCP/UDP source port. The filtering ruleset can become very complex because of it, and you can end up with flaws in the whole filtering scheme.

The *RPC* (Remote Procedure Call) services are very difficult to filter too. The associated servers listen at ports that are assigned randomly at system startup. The portmapper service maps initial calls to RPC services to the assigned service numbers. However, there is no such equivalent for a packet-filtering router. It becomes impossible to block these services completely because the router cannot be told on which ports the services reside (unless you block all UDP packets) because RPC services mostly use UDP. But if you block all UDP packets, you probably would block necessary services (DNS, for example). The question *becomes to block or not to block* RPCs.

You should get more information on packet filtering and associated problems. It's not the scope of this chapter to exhaust the subject, but packet filtering is a vital and important tool. It is very important to understand the problems it can present and how they can be addressed.

## Procuring a Firewall

After you've decided on the security policy, there are a number of issues to be considered in procuring a firewall. Standard steps to be taken are requirement definition, analysis, and design specification. The following sections describe some considerations, including minimal criteria for a firewall and whether to build or purchase a firewall.

## Needs Assessment

When the decision is made to use firewall technology to implement your organization's Web site security policy, the next step is to procure a firewall that provides the appropriate level of protection and cost-effectiveness. Ask these questions:

1. What features should a firewall have?
2. What would be considered effective protection?

Of course, by now you can entirely answer these questions with specifics but it is easy to assert that firewalls have the following features or attributes for which you should always look for:

- A firewall should be able to support a *deny all services, except for those specifically permitted* design policy. Even if you didn't read this chapter from the beginning, this should not be the policy to use! You must be able to *permit few* and still keep a sound level of security to your organization.
- A firewall should support your security policy, not force one.
- A firewall should be flexible. It should be able to be modulated to fit the needs of your company's security policy and be responsive to organizational changes.
- The firewall should contain advanced authentication measures or should be expandable to accommodate these authentications in the future.
- A firewall must employ filtering techniques that allow or disallow services to specified server systems as needed.
- The IP filtering language must be flexible, user-friendly to program, and capable of filtering as many attributes as possible, including source and destination IP addresses, protocol type, source and destination TCP/UDP ports, and inbound and outbound interfaces.
- A firewall should use proxy services for services such as FTP and telnet so that advanced authentication measures can be employed and centralized at the firewall. If services such as NNTP, X, HTTP, or Gopher are required, the firewall should contain the corresponding proxy services.
- A firewall should contain the capability to centralize SMTP access in order to reduce direct SMTP connections between site and remote systems. This will result in centralized handling of site e-mail.
- A firewall should accommodate public access to the site, such that public information servers can be protected by the firewall but can be segregated from site systems that do not require the public access.
- A firewall should contain the capability to concentrate and filter dial-in access.
- A firewall should contain mechanisms for logging traffic and suspicious activity, and should contain mechanisms for log reduction so those logs are readable and understandable.
- If the firewall requires an open operating system such as UNIX and NT, a secured version of the operating system should be part of the firewall, with other security tools as necessary to ensure firewall server integrity. The operating system should have all patches installed.
- A firewall should be developed in a manner so that its strength and correctness is verifiable. It should be simple in design so that it can be understood and maintained.
- A firewall and any corresponding operating system should be updated and maintained with patches and other bug fixes in a timely manner.

There are undoubtedly more issues and requirements, but many of them are specific to each site's own needs. A thorough requirements definition and high-level risk assessment will identify most issues and requirements; however, it should be emphasized that the Internet is a constantly changing network. New vulnerabilities can arise, and new services and enhancements to other services might represent potential difficulties for any firewall installation. Therefore, flexibility to adapt to changing needs is an important consideration.

# ***Buying a Firewall***

A number of organizations might have the capability to build a firewall for themselves. At the same time, there are a number of vendors offering a wide spectrum of services in firewall technology. Service can be as limited as providing the necessary hardware and software only, or as broad as providing services to develop security policy and risk assessments, security reviews, and security training.

Whether you buy or build your firewall, it must be restated that you should first develop a policy and related requirements before proceeding. If your organization is having difficulty developing a policy, you might need to contact a vendor who can assist you in this process.

If your organization has the in-house expertise to build a firewall, it might prove more cost-effective to do so. One of the advantages of building a firewall is that in-house personnel understand the specifics of the design and use of the firewall. This knowledge might not exist in-house with a vendor-supported firewall.

# ***Building a Firewall***

An in-house firewall can be expensive in terms of time required to build and document the firewall and the time required maintaining the firewall and adding features to it as required. These costs are sometimes not considered organizations sometimes make the mistake of counting only the costs for the equipment. If a true accounting is made for all costs associated with building a firewall, it could prove more economical to purchase a vendor firewall.

In deciding whether to purchase or build a firewall, answers to the following questions might help your organization decide whether it has the resources to build and operate a successful firewall:

- How will the firewall be tested?
- Who will verify that the firewall performs as expected?
- Who will perform general maintenance of the firewall, such as backups and repairs?
- Who will install updates to the firewall, such as new proxy servers, new patches, and other enhancements?
- Can security-related patches and problems be corrected in a timely manner?
- Who will perform user support and training?

Many vendors offer maintenance services along with firewall installation, so the organization should consider whether it has the internal resources needed.

# ***Setting It Up***

If you decide to build your firewall, make sure you respond to all of the preceding questions and that you indeed will be able to handle all the details of setting the firewall. Most importantly, make sure that your organization's upper management is 100-percent with you.

The following is an example of a firewall setup. Later on this chapter I give you an example of a firewall

installation, should you decide to purchase one instead of setting it up yourself. Hardware requirements and configuration will vary, of course, but if you follow the outlined steps you should be able to avoid lots of frustration and time-consuming surprises.

Also, make sure you have your firewall policy written up, understood, and on hand. When that is complete, write the following outlined steps on a board or notepad. They will be your roadmap in putting your firewall together:

- Select the hardware required.
- Install the necessary software (NOS, utilities, and so on).
- Connect and configure your machine on the network.
- Test it out.
- Add security (through firewalling software).
- Set up and configure the proxy server.

If your company is a medium size, I tried to complement the information to suit your needs with the following example of a company with 200 employees. Keep in mind: Far from being a sample firewall plan, this plan should be considered as a template to be modified as needed.

## Select the Hardware Required

Let's assume for example that I am setting up a firewall for my company, Vibes (Virtual Business Educational Services, in case you're wondering). For comparison reasons (and comparison reasons only!), consider Vibes to be a medium-sized company with 200 employees where all users have access to the Web and other services such as Telnet, FTP, Gopher, and SMTP.

The computer I will be using for the firewall is a 90Mhz Pentium with 16MB of RAM, a 540MB Linux partition, and a PPP connection to an Internet provider over a 28,800bps modem. To make the Linux box a firewall, I added an Ethernet network interface card and connected it to the company's LAN. All clients are running either Windows 95 (with SP1) or Windows NT Workstation 4.0 (with SP3 update).

## Install the Necessary Software

Now I have to set up my Linux box. I have to recompile the Linux kernel. In order to do that, I will have to issue a make config, where I will:

1. Turn on my network support and TCP/IP networking.
2. Turn off my IP forwarding (CONFIG\_IP\_FORWARD).
3. Turn on my IP firewalling.
4. Turn on IP accounting, which is not necessary but recommended, because we do want to institute security.
5. Turn on network device support.
6. Turn on my PPP and Ethernet support. If you're not using the same interface, you will have to make some adjustments here.

When done, I need to recompile and reinstall the kernel. I will then reboot the machine and watch the

interfaces showing up on the screen during my boot-up sequence (it should show up!). If not, I will need to review all of the above procedures, and even the machine itself if necessary. In doing so, I will watch for PCI and SCSI conflicts.

If everything works, it will be time to set up the system on the network.

## Connecting and Configuring the Computer on the Network

This part is crucial! In setting up the computer's network address, I need to keep in mind that I don't want the Internet to have access to my internal network (have you figured what kind of policy I'm using?). I am planning to use a fake network address. If you want to follow me on this one, a good C class you can use is 192.168.2.xxx, a dummy test domain.

I need to assign a real IP address to the serial port I will be using for my PPP connection, and assign 192.168.2.1 to the Ethernet card on my new domain JAVALITO. I will then assign a number in that domain to all the other computers in the protected network. It will then be time to test it out!

## Testing it

In order to test network connectivity, I will try to ping the Internet from JAVALITO. I want to make sure to try to ping a few other places that are not connected to my LAN. If it doesn't work, it will be an indication that I probably have set up my PPP incorrectly.

After I have a chance to ping out there, I will then try to ping a few hosts inside my own network. What I want to make sure here is that all of the computers on my internal network are able to ping each other. If not, it will not even be funny trying to continue with this setup until the problem is resolved, believe me!

As long as I determine that all of the computers are able to ping each other, they should also be able to ping JAVALITO. If not, I will have to go back to my previous step. One thing to remember is that I should try to ping 192.168.2.1, not the PPP address.

Lastly, I want to try to ping the PPP address of JAVALITO from inside my network. Of course, I should not be able to! If I can, this tells me that I have forgotten to turn off IP forwarding, and it will be time to recompile the kernel again. When I finish these tests, my basic firewall will be ready to go.

### Note:

You probably are thinking, why bother reconfiguring it, because I assigned my protected network to a dummy domain that consequently cannot get any packets routed to it? The reason is that by doing this I take the control away from my PPP provider and keep it in my own hands.

## Adding Security Through Firewalling Software

After I have my firewall set up, I will need to start "closing the doors," which at this point will still be quite open. Based on my policy, I will start turning off everything I don't need. At the top of my "turning off" list will be netstat, systat, tftp, bootp, Finger and rlogin. Once I turn off all of the services on my list,

I will try to Telnet the netstat port, which I shouldn't be able to get any output from. If I can, something is wrong.

At this point, my firewall will be up and running, but a firewall that doesn't allow anyone to come in or out is like a company that keeps its doors locked as part of a crime-prevention policy. It might be safe, but bad for business! By the same token, if a firewall is too restrictive, it can do as much harm as a wide-open firewall. With this in mind, applications, patches and software packages have been developed to make firewalls smarter and consequently more beneficial proxy servers, Socks, and so on.

Socks is one of several firewalling software packages out there, which is discussed in more details in the next section, discussing exclusively about proxies. TCP Wrapper is an application widely used as well, but as mentioned earlier in this chapter, it is not really a firewall utility so it is better to focus on Socks. Should you need additional information on TCP Wrapper, make sure to visit the FTP sites noted in that section.

## General Considerations When Installing a Firewall

Internet technology provides a cost effective global communications infrastructure that enables world-wide access for employees, customers, vendors, suppliers and key business partners. This is an important enhancement to collaborative information sharing, but it also exposes an organization's network to new risks and threats. How can an organization keep its resources and information protected from unauthorized network access, from both inside and outside the organization? Access control, a fundamental building block in any security policy, addresses this issue.

Access control protects an organization from security threats by specifying and enforcing what can go in and out of an organization's network. A key element of access control is an awareness of all underlying services and applications.

First generation packet filters were not aware of applications, nor could they handle UDP or dynamic protocols.

Second generation application proxies required a tremendous amount of CPU overhead, and were slow to provide support for new services appearing regularly on the Internet, such as multimedia services. Firewalls with stateful inspection technology, such as Check Point FireWall-1's, combined with a powerful object oriented approach, provide full application-layer awareness as well as quick and easy support of new Internet services. Firewall-1, for example, has over 160 pre-defined applications, services and protocols and the flexibility to specify and define custom services, providing a very comprehensive access control.

In addition to understanding the full state and context of a communication, FireWall-1 provides the ability for rules within a security policy to be enforced using a time parameter. This provides extensive granularity in access control allowing rules to be valid for specific hours/days/months/years. For example, an organization may decide to limit HTML or web traffic to the Internet during working hours, allowing access only during lunch time, after normal working hours and on weekends.

Another example is to disallow access to critical servers while system backups are being performed.

# Defining a Security Policy with a Firewall Product

Implementing access control parameters should be simple and straight forward with a well-defined graphical user interface such as that provided in most firewall products listed on chapter 14, "Types of Firewalls." In fact, all aspects of an organization's security policy are usually easy to be specified by using the GUI interfaces of these firewalls. Usually, all elements are specified using an object oriented approach. Once defined, these objects are used to define the security policy using a rules editor.

Although it varies from firewall to firewall product, each rule can be comprised of any combination of network objects, services, actions, and tracking mechanisms. In the example of Check Point's Firewall-1, once a rule is defined, FireWall-1 provides the ability to define which network enforcement points it should be distributed to across the network.

Supported platforms include UNIX and NT servers, and internetworking equipment (routers, switches, edge devices) from Check Point's many OPSEC Alliance partners. A distinct advantage of Check Point Firewall-1 is the ability to define an enterprise security policy once, distribute it to multiple access points throughout the network, and manage it from a single centralized console. Figure 7.7 shows a screenshot of Firewall-1 security policy setup.

Figure 7.8 shows four separate components of the Graphical User Interface (GUI) for Firewall-1. These components are as follows:

- Rules Base Editor - (upper right hand) This is the main editor to define the security policy to be installed on the firewall.
- Network Objects Manager - (upper left hand) This tool is used to create hosts, gateways, networks, domain, routers, groups of objects.
- Users Manager - (lower left) This tool is used to create Users, Groups of Users, and Templates for the creation of Users.
- System Status View - (lower right) This tool monitors the current status of the firewalls.

Figure 7.9 shows the screen that will appear once you select the gateway.

Figure 7.10 shows the host properties screen of Firewall-1 and figure 7.11 shows the users management screen. These screenshots give you an idea of what to expect on a top-of-the-line firewall product. Keep it in mind when shopping for a firewall. Needless to say, Check Point's product should be strongly considered.

## Adminstrating a Firewall

Firewall administration is a critical job role and should be afforded as much time as possible. In small organizations, it might require less than a full-time position, but it should take precedence over other duties. The cost of a firewall should include the cost of administrating the firewall administration should never be shortchanged.

# ***Management Expertise***

As described at the beginning of this chapter, there are many ways to break into a system through the Internet. Therefore, the need for highly trained, quality, full-time server system administrators is clear. But there are also indications that this need is not being met satisfactorily in a way that identifies, protects, and prevents such incidents from happening. Many system managers are part-time at best and do not upgrade systems with patches and bug fixes as they become available.

Firewall management expertise is a highly critical job role because a firewall can only be as effective as its administration. If the firewall is not maintained properly, it might become insecure and permit break-ins while providing the illusion that the site is still secure. A site's security policy should clearly reflect the importance of strong firewall administration. Management should demonstrate its commitment to this importance in terms of full-time personnel, proper funding for procurement and maintenance, and other necessary resources.

## ***System Administration***

A firewall is not an excuse to pay less attention to site system administration. It is, in fact, the opposite: If a firewall is penetrated, a poorly administered site could be wide open to intrusions and resultant damage. A firewall in no way reduces the need for highly skilled system administration.

At the same time, a firewall can permit a site to be proactive in its system administration as opposed to reactive. Because the firewall provides a barrier, sites can spend more time on system-administration duties and less time reacting to incidents and damage control. It is recommended that sites do the following:

- Standardize operating system versions and software to make installation of patches and security fixes more manageable.
- Institute a program for efficient, site-wide installation of patches and new software.
- Use services to assist in centralizing system administration if it will result in better administration and better security.
- Perform periodic scans and checks of server systems to detect common vulnerabilities and errors in configuration, and to ensure that a communications pathway exists between system administrators and firewall/site security administrators to alert the site about new security problems, alerts, patches, and other security-related information.
- Finally, ask yourself: What kind of firewall do I need? There is no correct answer. A security plan chosen by company A may not be suitable for company B. Here are few suggested scenarios.

## **Circuit-Level Gateways and Packet Filters**

For a company which relies greatly on outgoing access capabilities, such as educational companies, universities, etc , it is recommended that circuit-level gateways and packet filters be used. This assumes that the departments of this company trust their internal users. If the installation will restrict outsiders to accessing only a Web server, outside of the firewall, blocking any external connections from the internal



and protected network, the department might not need anything more.

## Packet Filtering

The same model will suit a mid-size company relying heavily on the Internet, such as ISPs, Web hosting, etc., but the policy will be contrary to the example above since more Internet users will be accessing the site than the site accessing the Internet. Wide access can be granted to the Web/Internet server outside of the firewall. Protected network users would have to Telnet to the Internet/Web server, from inside the company, just like everyone else outside of the firewall.

## Application Gateways

Larger companies or those where Internet users are offered access to specific services and shares inside the protected network will need to have a different setup. In this case, I would suggest firewall packages like CheckPoint, or at least an application gateway. It would be advisable to implement CERT's recommendation of an additional router to filter and block all packets whose addresses are originated from inside the protected network. This two-router solution is not complicated to deploy, and is very cost-effective when you consider that a larger company would be exposed to spoofing by allowing all the many employees it has throughout the country to have access to its Web server and internal network.

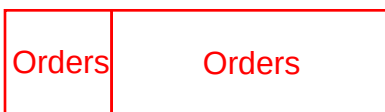
When implementing two routers, you should purchase them from different companies (that is, choose two different brands). It might sound like nonsense, but if a hacker is able to break into one router due to a bug or a back door on the router's code, the second router will not have the same codes. Even though the firewall will no longer be transparent, which will require users to log on to it, the site will be protected, monitored, and safe.

The typical firewall for such a company is illustrated on Figure 7.12. In this case, the two routers create a package-filtering firewall while the bastion gateway functions as an application-gateway firewall.

## IP-Level Filtering

In the case of a smaller-sized company, the IP-level filtering might be the most appropriate versus other types of filtering. This model enables each type of client and service basically to be supported within the internal network. No modifications or special client software would be necessary.

The access through the IP-level filtering firewall will be totally transparent for the user and the application. The existing router can be utilized for the implementation of the IP-level filtering. There will be no need to buy an expensive UNIX host. However, small companies can reinforce its Internet server security by implementing similar solutions used by a larger company without a need for the application gateway.



|       |          |
|-------|----------|
| Comme | Comments |
|-------|----------|

---

[COMPUTING MCGRAW-HILL](#) | [Beta Books](#) | [Contact Us](#) | [Order Information](#) | [Online Catalog](#)

---

HTML conversions by [Mega Space](#).

This page updated on December 05, 1997 by [Webmaster](#).

[Computing McGraw-Hill](#) is an imprint of the [McGraw-Hill Professional Book Group](#).

Copyright ©1997 [The McGraw-Hill Companies, Inc.](#) All Rights Reserved.

Any use is subject to the rules stated in the [Terms of Use](#).

|        |        |
|--------|--------|
| Orders | Orders |
|--------|--------|

|      |         |
|------|---------|
| Back | Forward |
|------|---------|

|          |          |
|----------|----------|
| Comments | Comments |
|----------|----------|

© 1997 The McGraw-Hill Companies, Inc. All rights reserved.

Any use of this Beta Book is subject to the rules stated in the [Terms of Use](#).

---

## Chapter 8

# *How Vulnerable Are Internet Services?*

The implementation of Internet services must be carefully considered due to its vulnerabilities and threats. This chapter lists some of the most common implemented services and discusses the risks associated with each one of them.

## Protecting and Configuring Vulnerable Services

In this section, let's take a look at the industry standard protocols and services, its characteristics and how they interact with the Internet and firewall so we can be aware of security threats and countermeasures as well as a configuration checklist, with loopholes to watch for and security issues.

## *Electronic Mail Security Threats*

Electronic mail (E-mail) is a wonderful tool to have on the Internet, but it brings threats to your privacy and security. This section discusses some of these threats, such as e-mail bombing and spamming, as well as the risks of downloading certain attachments.

One of the main weaknesses of e-mail messages is that not always it can be traced. The Reuters, awhile ago, published an article about President Clinton receiving a death threat over the Internet via e-mail. According to the article, the e-mail message was originated at a Taiwan university and contained the message

"President Clinton, when you are out for a visit, we will assassinate you."

The United States asked Taiwan officials to investigate the incident, but the director of the university computer center concluded that there was no way to find a record of the person logging in and out on the Internet and sending the message to President Clinton. Thus, you should be aware of e-mail threats and what you can do to prevent against these pitfalls.

You can be threaten by anyone using an anonymous e-mail, and you won't be able to track him or her down. Take this other example, of Jonathan Littman, one of the few journalists covering the computer underground. When Kevin Mitnick, was arrested, Littman had become the uber-hacker's inside, to the extend of even writing a book, entitled "The Fugitive Game." The problem is that on the book he was sympathetic to Mitnick, and ended up receiving some retaliation for some hackers, which sent him several e-mail threats, vowed through anonymous messages.

E-mail threats also includes people scanning your messages in search of valuable information, such as credit card, social security numbers or systems authentication information? When an e-mail message travels through the Internet it can be exposed to little programs that automatically will scan the mail feed into a computer, looking for specific information, just like you do in your mail program when you want to locate a particular message stored in one of your message folders.

A good preventive measure to this kind of attack is through message encryption. As discussed on chapter 3, "Cryptography, Is It Enough," encryption does hacking much more difficult. Also, there are lots of encryption tools out there, such as Pretty Good Privacy (PGP) and digital signatures to aid you on this process. You should encrypt and sign all your e-mail messages.

**Tip:**

For information on protecting electronic information, check out "An Attorney's Guide To Protecting, Discovering, and Producing Electronic Information" by Michael Patrick (phone 1-800-341-7874 x-310).

**Note:**

If you would like more information on e-mail bombing and spamming, check out Byron Palmer's Web page at URL [http://mwir.lanl.gov:8080/E-mail\\_Spamming.html](http://mwir.lanl.gov:8080/E-mail_Spamming.html)

## Simple Mail Transfer Protocol (SMTP)

Have you heard about e-mail bombing? This is a form of stalking, an anonymous type of harassment to which you can't reply back to the sender. E-mail bombing is illegal, but hard to track, because of the anonymous ways e-mail can be sent, usually consisting of sending large amount of messages, from hundreds to thousands of e-mail messages, to a single e-mail address, usually generating a denial-of-service on the mail server.

But don't confuse e-mail bombing with spamming. E-mail *bombing* is characterized by abusers repeatedly sending numerous copies of the same e-mail message to a particular address, whereas e-mail *spamming* is a variant of bombing; it refers to sending the same e-mail to hundreds or thousands of users (or to lists that expand to that many users). E-mail spamming can be made worse if recipients reply to the e-mail, causing all the original addressees to receive the reply. Spamming also may occur innocently, as a result of someone sending a message to a mailing list without realizing that the list explodes to thousands of users, or as a result of an incorrectly set-up responder message. If the identity of the account sending the message is altered, then e-mail bombing or spamming is being combined with "spoofing," which makes it almost impossible to track the author and the origin of the message. Later on this chapter there is a section on spoofing, make sure to read it.

**Tip:**

A good source of information about e-mail spoofing is at the URL [ftp://info.cert.org/pub/tech\\_tips/email\\_spoofing](ftp://info.cert.org/pub/tech_tips/email_spoofing).

As mentioned in the chapter above, the large amount of e-mails coming in to a server, as a result of e-mail bombing and spamming can generate a denial-of-service (where the server denies to honor a request or a task, to the extreme of freezing up) on the server, through loss of network connectivity, system crashes, or even failure of a service (where the ability to execute that service fails on the server) because of:

- overloaded network connections
- used up system resources
- filled up disk as a result of multiple postings and syslog entries

## Preventing against E-mail Attacks

It is very important that you are able to detect e-mail bombing or spamming as soon as possible. One of the signs your system will present when under attack is sluggishness. If e-mail is slow or is not being sent or received, it could be that your mail server is either trying to process a large number of messages, or already has suffered a denial-of-service, as mentioned above.

If you are experiencing such a condition in your server, I recommend you do:

- Identify the source of the e-mail by checking the headers, and immediately reconfigure your firewall (or router) to block incoming packets from that address. Be careful before assuming that the author of the attack is the person showing on the header of the message, as many times the name appearing there is just an alias, in an attempt to hide the true identity.
- If your e-mail service is through an Internet Service Provider (ISP), let them know about the bombing or spamming incident

so that they can reconfigure their router or firewall, to prevent messages coming from the address of origin.

- Contact the Computer Emergency Response Team (CERT) at **cert@cert.org** about the attack so that they can track the incidents. The CERT Coordination Center charter is to work with the Internet community to facilitate its response to computer security events involving Internet hosts, to take proactive steps to raise the community's awareness of computer security issues, and to conduct research targeted at improving the security of existing systems.

There is no way to block e-mail bombing and spamming. However, there are a few things you can do to protect yourself and decrease the likelihood of a bombing or spamming attack. One, you should keep your e-mail software up to date at all times. Two, make sure you maintain the updates, patches, and bug fixes that are released by your e-mail developer. The third thing is a little more technical. You could develop a tool that would check for and alert you to incoming messages that originate from the same user or same site in a short span of time. You then could block these connections at the router level.

For example, once you identify from where this messages are coming from, the sites domain (spuky@badboy.com, for example) you can go to your firewall and block, or deny, any messages coming from that site. You can even re-direct it to a wastebasket directory where it will be periodically deleted. You will probably not be able to identify the author of these messages but you can at least stop receiving them by blocking them before they hit your mailbox.

One alternative to keep in mind is that, if you have only one or two e-mail servers, make sure to set up your firewall to allow only SMTP connections coming from the Internet to your e-mail server. From there you will have to block the SMTP port not to allow connections arriving directly at the user's computers.

The way you do this will vary from firewall to firewall software. If you are using an router as firewall, you will have to insert a line on its configuration file, denying connections to the SMTP port. By blocking access to your SMTP port you will prevent the injection of spamming messages through it.

You can block the SMTP port by turning off your mailers SMTP daemon mode and run it out of inetd instead. If you combine this with running smap from the TIS Firewall Toolkit, the configuration will look like this:

At the /etc/inetd.conf:

```
smtp      stream  tcp      nowait  root    /usr/local/etc/tcpd    smap
```

At the /etc/hosts.allow:

```
smap : ALL
```

At /etc/hosts.deny:

```
smap : spammer.com .spammer.com 128.xxx.000.0
```

at the /usr/local/etc/netperm-table:

```
smap, smapd:          userid 32
```

```
smap, smapd:          directory /var/spool/smap
```

```
smapd:                executable /usr/local/libexec/smapd
```

```
smapd:                sendmail /usr/sbin/sendmail
```

You can use the above example as a boilerplate, as the paths will vary according to your environment as well as the site(s) you're blocking. This should suffice to keep e-mail spamming and bombing coming from spammer.com or anyone in the IP range of 128.xxx.000.0 from accessing your SMTP server. Now, watch your server! This technique could overload your server as it will generate a process for every incoming mail message. If your server already works at more than 30% of its capacity, you may want

to try a different technique discussed here.

**Note:**

You can try to block spamming by using smap. According to Craig Hagan (hagan@cih.com), spammers often use third-party relaying to distribute spam via an intermediary party's mailer, so Hagan proposes a routine, which you can review and download from URL <http://www.cih.com/~hagan/smap-hacks/> to prevent your mailer from being misused by the relaying mailer.

You can also use sendmail to block spamming and bombing. Axel Zinser (azin0999@rz.uni-hildesheim.de) has developed patches for blocking spam with sendmail versions 8.6.12, 8.7.3 or 8.8.2. For more information, check the URL <http://spam.abuse.net/spam/tools/mailblock.html>.

**Tip:**

Check CERT for additional information on filtering SMTP connections in your firewall, at URL [ftp://info.cert.org/pub/tech\\_tips/packet\\_filtering](ftp://info.cert.org/pub/tech_tips/packet_filtering).

## Be Careful With E-Mail Attachments

Have you ever thought about the potential danger of e-mail attachments? The majority of the mail packages, such as MS Exchange and cc:Mail have a setting in which you can specify if you want attachments to go as separate files , or to be encapsulated. The risk with these attachments is that they can contain various threats, from viruses and malicious macros to small Trojan Horses applets.

According to Integralis, the developers of MIMESweeper e-mail security (<http://www.mimesweeper.integralis.com/>), viruses can be inserted into email attachments, just as part of the header or even in the body. Some of the most threatening and successful current viruses and logic bomb codes are document-based and as you probably know, most email attachments are also documents. Therefore, e-mail attachments becomes one of the easiest way to get a virus into your computer or company.

In order to protect yourself against such a threat you must be very careful with opening attachments, as they are the carriers of the threat. Make sure you know the origin of it and that you can trust the attachment is clean and free of bugs! If you can't, then you must use an e-mail virus detector to scan the messages you receive, such as MIMESweeper, .

An especial attention should be given to encoded and ZIPped file attachments, as many times they are skipped or not supported by anti-virus packages. This attachments could contain viruses and macro bombs.

There are viruses that can gain access to your computer via the attachments you download or open from your mail that could apparently damage your computer. Even though the media portraits that viruses can damage your computer hardware, I haven't yet seeing a single one capable of doing so. Nevertheless, there are viruses that can make your hard drive behave as it was faulty.

For instance, if you were to activate a virus, downloaded along with an attachment, known as Rainbow, this virus would alter the partition table located in the Master Boot Record of your hard drive in such a way that, if you attempt to boot from a clean uninfected system disk with MS-DOS 5.x or 6.x, the machine would simply hang.

The most notorious virus attached to e-mail messages of 1997 is the so called AOL4FREE.COM, which actually is a Trojan Horse. This Trojan is a simple .BAT file created to issue a DOS command (DELTREE) to delete the complete file directory tree on the hard drive. Although you can not activate this Trojan by simply reading your e-mail, you must be very careful when dealing with attachments.

Another famous e-mail attached virus that appeared on the Net around February of 1997 was ShareFun.A. Although a macro virus, this one has a new feature: once activated, there is a 25% chance that it will launch MS Mail and attach itself to a newly self-created email message and then grab 3 random mail addresses from your Personal Address Book (PAB) file, set the Subject line to read "You have GOT to read this!" and send it on it's way.

In order to protect yourself against e-mail attachments make sure you know the origin of it and that you can trust the attachment is

clean and free of bugs! If you can't, then you must use an e-mail virus detector to scan the messages you receive, such as MIMESweeper, from Integralis.

E-mail anti-viruses packages can scan e-mail messages as they come in from the host mail system. By using a recursive disassembly, these applications can completely open your message and any attachments so that anti-virus tools can check for viruses embedded within the data.

As for macro viruses, make sure to download the latest version of Word/Excel macro anti-viruses, SCANPROT.DOT from Microsoft or even third-parties such as Datafellows, at URL://www.datafellows.com. Note that by simply installing SCANPROT.DOT will not protect you from being infected by macro virus attached to your e-mails. If you open a Word or Excel document simply by clicking with your mouse over the attachment, SCANPROT will not be started. You must download the file, launch Word or Excel and open the file from within the application.

If the security of your SMTP connections and corporate messages traveling through it is really important, I recommend you to consider using Riordan's Internet Privacy Enhanced Mail (RIPEM), which it is a still-to-be completed but practical implementation of Privacy Enhanced Mail (PEM).

PEM is a standard for allowing the transfer of encrypted electronic mail generated over a long period of time by a working group of specialists. Note that RIPEM is not really a complete implementation of PEM. RIPEM specifies certificates for authenticating keys, and RIPEM does not handle those yet. The addition of key authentication is planned for the near future, as well as for the Macintosh version, which are different from the PC version due to their distinct operating system. RIPEM provides your SMTP mail with the security facilities provided by PEM, which are:

- **Disclosure protection**, which is optional and protects the contents of messages from unauthorized disclosure.
- **Originator authenticity**, which allows digital signature and reliability of a message to be verified.
- **Message integrity measures**, which assures that the message has not being modified during the transmission, and
- **Non-repudiation of origin** which allows for the verification of the identity of the original sender of the message.

**Note:**

For more information on RIPEM, and if you want to download a copy of it, check the FTP site at URL <ftp://ftp.rsa.com/rsaref/>. Note that there are restrictions for downloading RIPEM, as it uses the RSAREF library of cryptographic routines, which is considered munitions and thus is export-restricted from distribution without an export license to persons who are not citizens or permanent residents in the U.S or Canada. Thus, I strongly recommend you to read the frequently asked questions for RIPEM at URL <http://www.cs.indiana.edu/ripem/ripem-faq>.

You can use RIPEM with popular mailers such as Berkeley, mush, Elm, and MH. Code also is included in elisp to allow the easy use of RIPEM inside GNU Emacs. Post your interfaces or improvements for RIPEM to the newsgroup on USENET, **alt.security.ripem**.

Zimmermann's Pretty Good Privacy (PGP), is another product you can use to encrypt your SMTP messages. However, unlike RIPEM, PGP tries to approach the issue of trustworthiness, but as I understand it, it does so without respect to any enunciated criteria or policy. Thus the question remains: Can you trust someone you're with whom you are interacting through e-mail, by signing a contract or something similar (using digital signatures), just because he's authenticated over PGP or RIPEM?

## ***Post Office Protocol (POP)***

As an Internet standard, POP (Post Office Protocol) defines the means of accessing and downloading electronic mail from a server. POP clients use the SMTP protocol to SEND messages, POP is only used to retrieve messages. POP version 2, or POP2 (or POP3) are standards wide in use, especially POP3, which added some new functionality to the interface. POP is also a TCP/IP based protocol, meaning you need a network connection between client and host.

POP2 or POP3 clients are available from a wide variety of sources on the Internet for MSDOS, Windows, OS/2, UNIX, Macintosh, and several other platforms. As you probably already know, POP clients look and feel just like PC-based e-mail packages and require no access to the host (server) other than a mailbox and mailbox password.

With POP, mail is delivered to a shared server, which then is retrieved by an user that connects to the server and downloads all of

the pending mail to the "client" machine. Thereafter, all mail processing is local to the client machine.

But you must keep in mind that when you are dealing with POP configuration you ultimately are dealing with private information coming and going through it. You are dealing with issues such confidentiality, integrity and liabilities! Thus, I recommend you not to allow your users to transfer mail over the Internet through a POP, because it can reveal passwords and the messages are totally unprotected. If they must transfer it, then implement packet filtering. You might be able to implement some proxy too, but it will require some minor coding.

Recently, CERT Advisory CA-97.09 (August 27, 1997), reported on a vulnerability with POP and Internet Message Access Protocol (IMAP). According to CERT, some versions of the University of Washington's implementation of the IMAP and POP has a security hole that allows remote users to obtain unauthorized root access without even having access to an account on the system.

CERT/CC team recommends installing a patch if one is available or upgrading to IMAP4rev1. Until you can do so, CERT recommends you to disable the IMAP and POP services at your site.

**Tip:**

Should you need to update to IMAP4rev1, you can download it from the University of Washington FTP server at URL <ftp://ftp.cac.washington.edu/mail/imap.tar.Z>. Note that the checksums change when files are updated.

If you are not able to temporarily disable the POP and IMAP services, then try to limit access to the vulnerable services to machines in your local network. This can be done by installing the tcp\_wrappers, since POP is launched out of inetd.conf, for loggins and access control. This doesn't mean that your POP is safe now, and you still have to do run the fix, hopefully already available by the publishing of this book, or upgrade to IMAP4ver1. Additionally, you should consider filtering connections at the firewall to minimize the impact of unwanted connections.

**Note:**

If you need access to the tcp\_wrappers tool, you can download it from CERT's FTP server at URL [ftp://info.cert.org/pub/tools/tcp\\_wrappers/tcp\\_wrappers\\_7.5.tar.gz](ftp://info.cert.org/pub/tools/tcp_wrappers/tcp_wrappers_7.5.tar.gz)

The BorderWare firewall is an example of a product that runs all standard Internet servers including a full function electronic mail server with POP and SMTP support. But BorderWare is not the only one, check chapter 14, "Types of Firewall," for the complete list.

## ***Multimedia Internet Mail Extensions (MIME)***

MIME is an acronym for Multipurpose Internet Mail Extensions. The standard for attaching non-text files to standard Internet mail messages. Unfortunately, MIME is not secure. Thus, RSA developed S/MIME, which is a specification for secure MIME by offering authentication (using digital signatures) and privacy (using encryption).

S/MIME, PGP, and PEM are similar, as they specify methods for securing your electronic mail. However, PGP can be thought of as both a specification and an application as it relies on users to exchange keys and establish trust in each other. S/MIME, on the other hand, utilizes hierarchies in which the roles of the user and the certifier are formalized, which makes S/MIME more secure and more scaleable than PGP implementations.

If we were to compare PEM with S/MIME, we'll need to take in consideration that PEM is an early standard for securing e-mail that specified a message format and a hierarchy structure. The PEM message format is based on 7-bit text messages, whereas S/MIME is designed to work with MIME binary attachments as well as text. The guidelines for hierarchies are also more flexible in S/MIME. This should allow for both easy set-up for small workgroups that don't need to be part of an all-encompassing hierarchy, and an easy path to move workgroups to the hierarchy that best suits their needs.



**Note:**

For more information about RSA's S/MIME, check their URL at <http://www.rsa.com/smime/html/faq.html#gnrl.1>.

Now, one way to have more control over your SMTP mail is to tunnel them to a specific server where they can be screened. You can easily do this by setting up an HTML email form and using the "Mailto" function. You would enter a line code it in HTML as

```
<A HREF="mailto:user@hisdomain.com">user@hisdomain.com</A>
```

The *safedude@murphylaw.com* eventually will be replaced by an Internet address. Every time an user clicks on the email anchor, a special form pops-up. The user then writes his message and sends it to you.

However, there are many other options, in many different script languages. It all will depend on how much you want to invest on it, in time and effort, and the resources you have available.

To create an email comment form, you will need to create a form which sends mail to you from any browser that supports forms. For UNIX server, there is a very flexible CGI script, *cgimail*, which can be downloaded from MIT's Web site. I have not seen any other tool for this purpose with such a level of flexibility. It is also very easy to install and use.

Since *cgimail* requires an ASCII form, it can be later emailed, which allows users with disabilities to access it. If you want to download it, check the *mit-dcns-cgi* at the URL: <http://web.mit.edu/wwwdev/www/dist/mit-dcns-cgi.html>.

If you rather work with ANSI C, there is very simple email form package called Simple CGI Email Handler, which I strong recommend. It is based on the *post\_query.c* code provided with the NCSA *httpd* 1.1 package, released to the public domain.

You should be aware of AIX, which definitely is vulnerable to it. The SunOS 4.1.3 does not allow these escape sequences, unless mail is being run from an actual terminal. With version 2.1, you don't need to be concerned about it as the tilde escapes were replaced with spaces.

If you are interested on this script, you can download it from the URL: <http://www.boutell.com/email/>.

If you like Perl, there is another email form package called the "Web Mailto Gateway," developed by Doug Stevenson ([doug+@osu.edu](mailto:doug+@osu.edu)). The following source code can be found at URL: [http://www.mps.ohio-state.edu/mailto/mailto\\_info.html](http://www.mps.ohio-state.edu/mailto/mailto_info.html).

```
#!/usr/local/bin/perl
```

```
#
```

```
# Doug's WWW Mail Gateway 2.2
```

```
# 5/95
```

```
# All material here is Copyright 1995 Doug Stevenson.
```

```
#
```

```
# Use this script as a front end to mail in your HTML. Not every browser
```

```
# supports the mailto: URLs, so this is the next best thing. If you
```

```
# use this script, please leave credits to myself intact! :) You can
```

```
# modify it all you want, though.

#

# Documentation at:

#   http://www-bprc.mps.ohio-state.edu/mailto/mailto_info.html

#

# Configurable items are just below.  Also pay special attention to

# GET method arguments that this script accepts to specify defaults

# for some fields.

#

# I didn't exactly follow the RFCs on mail headers when I wrote this,

# so please send all flames my way if it breaks your mail client!!

# Also, you'll need cgi-lib.pl for the GET and POST parsing.  I use

# version 1.7.

#

# Requires cgi-lib.pl which can be found at

#   http://www.bio.cam.ac.uk/web/form.html

#

# PLEASE: Use this script freely, but leave credits to myself!!  It's

#   common decency!
```

#

#####

#

# Changes from 1.1 to 1.2:

#

# A common modification to the script for others to make was to allow

# only a certain few mail addresses to be sent to. I changed the WWW

# Mail Gateway to allow only those mail addresses in the list @addrs

# to be mailed to - they are placed in a HTML <SELECT> list, with either

# the selected option being either the first one or the one that matches

# the "to" CGI variable. Thanks to Mathias Koerber

# <Mathias.Koerber@swi.com.sg> for this suggestion.

#

# Also made one minor fix.

#

#####

#

# Changes from 1.2 to 1.3:

#

# Enhancing the enhancements from 1.2. You can now specify a real name

```
# or some kind of identifier to go with the real mail address.  This

# infomation gets put in the %addrs associative array, either explicitly
# defined, or read from a file.  Read the information HTML for instructions
# on how to set this up.  Also, real mail addresses may hidden from the
# user.  Undefine or set to zero the variable $expose_address below.

#

#####

#

# Changes from 1.3 to 1.4

#

# The next URL to be fetched after the mail is sent can be specified with
# the cgi varaible 'nexturl'.

#

# Fixed some stupid HTML mistake.

#

# Force user to enter something for the username on 'Your Email:' tag,
# if identd didn't get a username.

#

# Added Cc: field, only when %addrs is not being used.

#
```

#####

#

# Quickie patch to 1.41

#

# Added <PRE>formatted part to header entry to make it look nice and fixed a

# typo.

#

#####

#

# Version 2.0 changes

#

# ALL cgi variables (except those reserved for mail info) are logged

# at then end of the mail received. You can put forms, hidden data,

# or whatever you want, and the info for each variable will get logged.

#

# Cleaned up a lot of spare code.

#

# IP addresses are now correctly logged instead of just hostnames.

#

```
# Made source retrieval optional.
```

```
#
```

```
#####
```

```
#
```

```
# Changes from 2.0 to 2.1
```

```
#
```

```
# Fixed stupid HTML error for an obscure case.  Probably never noticed.
```

```
#
```

```
# Reported keys are no longer reported in an apparently random order; they
```

```
# are listed in the order they were received.  That was a function of perl
```

```
# hashes...changed to a list operation instead.
```

```
#
```

```
#####
```

```
#
```

```
# Changes from 2.1 to 2.2
```

```
#
```

```
# Added all kinds of robust error checking and reporting.  Should be
```

```
# easier to diagnose problems from the user end.
```

```
#
```

```
# New suggested sendmail flag -oi to keep sendmail from ending mail
```

```
# input on line containing . only.

#

# Added support for setting the "real" From address in the first line

# of the mail header using the -f sendmail switch.  This may or may not

# be what you want, depending on the application of the script.  This is

# useful for listservers that use that information for identification

# purposes or whatever.  This is NOT useful if you're concerned about

# the security of your script for public usage.  Your mileage will vary,

# please read the sendmail manual about the -f switch.

#   Thanks to Jeff Lawrence (jlaw@irus.rru.uwo.ca) for figuring this

#   one out.

#

#####

#

# Doug Stevenson

# doug+@osu.edu

#####

# Configurable options

#####

# whether or not to actually allow mail to be sent -- for testing purposes
```

```
$active = 1;

# Logging flag.  Logs on POST method when mail is sent.

$logging = 1;

$logfile = '/usr/local/WWW/etc/mailto_log';

# Physical script location.  Define ONLY if you wish to make your version

# of this source code available with GET method and the suffix '?source'

# on the url.

$script_loc = '/usr/local/WWW/cgi-bin/mailto.pl';

# physical location of your cgi-lib.pl

$cgi_lib = '/usr/local/WWW/cgi-bin/cgi-lib.pl';

# http script location

$script_http = 'http://www-bprc.mps.ohio-state.edu/cgi-bin/mailto.pl';

# Path to sendmail and its flags.  Use the first commented version and

# define $listserver = 1 if you want the gateway to be used for listserver

# subscriptions -- the -f switch might be necessary to get this to work

# correctly.

#

# sendmail options:

#    -n  no aliasing
```



```
# -t read message for "To:"

# -oi don't terminate message on line containing '.' alone

#$sendmail = "/usr/lib/sendmail -t -n -oi -f"; $listserver = 1;

$sendmail = "/usr/lib/sendmail -t -n -oi";

# set to 1 if you want the real addresses to be exposed from %addrs

#$expose_address = 1;

# Uncomment one of the below chunks of code to implement restricted mail

# List of address to allow ONLY - gets put in a HTML SELECT type menu.

#

#%addrs = ("Doug - main address", "doug+@osu.edu",

#         "Doug at BPRC", "doug@polarmet1.mps.ohio-state.edu",

#         "Doug at CIS", "stevenso@cis.ohio-state.edu",

#         "Doug at the calc lab", "dstevens@mathserver.mps.ohio-state.edu",

#         "Doug at Magnus", "dmsteven@magnus.acs.ohio-state.edu");

# If you don't want the actual mail addresses to be visible by people

# who view source, or you don't want to mess with the source, read them

# from $mailto_addrs:

#

#$mailto_addrs = '/usr/local/www/etc/mailto_addrs';

#open(ADDRS,$mailto_addrs);
```

```

#while(<ADDRS>) {

#    ($name,$address) = /^(.+)[ \t]+([^\ ]+)\n$/;

#    $name =~ s/[ \t]*$//;

#    $addrs{$name} = $address;

#}

# version

$version = '2.2';

#####

# end of configurable options

#####

#####

# source is self-contained

#####

if ($ENV{'QUERY_STRING'} eq 'source' && defined($script_loc)) {

    print "Content-Type: text/plain\n\n";

    open(SOURCE, $script_loc) ||

        &InternalError('Could not open file containing source code');

    print <SOURCE>;

    close(SOURCE);

    exit(0);

```

}

require \$cgi\_lib;

&amp;ReadParse();

#####

# method GET implies that we want to be given a FORM to fill out for mail

#####

if (\$ENV{'REQUEST\_METHOD'} eq 'GET') {

# try to get as much info as possible for fields

# To: comes from \$in{'to'}

# Cc: comes from \$in{'cc'}

# From: comes from REMOTE\_IDENT@REMOTE\_HOST || \$in{'from'} || REMOTE\_USER

# Subject: comes from \$in{'sub'}

# body comes from \$in{'body'}

\$destaddr = \$in{'to'};

\$cc = \$in{'cc'};

\$subject = \$in{'sub'};

\$body = \$in{'body'};

\$nexturl = \$in{'nexturl'};

if (\$in{'from'}) {

```
$fromaddr = $in{'from'};

}

# this is for NetScape pre-1.0 beta users - probably obsolete code

elsif ($ENV{'REMOTE_USER'}) {

    $fromaddr = $ENV{'REMOTE_USER'};

}

# this is for Lynx users, or any HTTP/1.0 client giving From header info

elsif ($ENV{'HTTP_FROM'}) {

    $fromaddr = $ENV{'HTTP_FROM'};

}

# if all else fails, make a guess

else {

    $fromaddr = "$ENV{'REMOTE_IDENT'}\@$ENV{'REMOTE_HOST'}";

}

# Convert multiple bodies (separated by \0 according to CGI spec)

# into one big body

$body =~ s/\0//;

# Make a list of authorized addresses if %addrs exists.

if (%addrs) {

    $selections = '<SELECT NAME="to">';
```

```

foreach (sort keys %addrs) {

    if ($in{'to'} eq $addrs{$_}) {

        $selections .= "<OPTION SELECTED>$_";

    }

    else {

        $selections .= "<OPTION>$_";

    }

    if ($expose_address) {

        $selections .= " &lt;$addrs{$_}>";

    }

}

$selections .= "</SELECT>\n";

}

```

```
# give them the form
```

```
print &PrintHeader();
```

```
print <<EOH;
```

```
<HTML><HEAD><TITLE>Doug\'s WWW Mail Gateway $version</TITLE></HEAD>
```

```
<BODY><H1><IMG SRC="http://www-bprc.mps.ohio-state.edu/pics/mail2.gif" ALT="">
```

```
The WWW Mail Gateway $version</H1>
```

```
<P>The <B>To</B>: field should contain the <B>full</B> Email address
```

that you want to mail to. The **Your Email**: field needs to contain your mail address so replies go to the right place. Type your message into the text area below. If the **To**: field is invalid, or the mail bounces for some reason, you will receive notification if **Your Email**: is set correctly. *If **Your Email**: is set incorrectly, all bounced mail will be sent to the bit bucket.*

```
<FORM ACTION="$script_http" METHOD=POST>
```

```
EOH
```

```
;
```

```
print "<P><PRE>          <B>To</B>: ";
```

```
# give the selections if set, or INPUT if not
```

```
if ($selections) {
```

```
    print $selections;
```

```
}
```

```
else {
```

```
    print "<INPUT VALUE=\"\$destaddr\" SIZE=40 NAME=\"to\">\n";
```

```
    print "          <B>Cc</B>: <INPUT VALUE=\"\$cc\" SIZE=40 NAME=\"cc\">\n";
```

```
}
```

```
print <<EOH;
```

<B>Your Name</B>: <INPUT VALUE="\$fromname" SIZE=40 NAME="name">

<B>Your Email</B>: <INPUT VALUE="\$fromaddr" SIZE=40 NAME="from">

<B>Subject</B>: <INPUT VALUE="\$subject" SIZE=40 NAME="sub"></PRE>

<INPUT TYPE="submit" VALUE="Send the mail">

<INPUT TYPE="reset" VALUE="Start over"><BR>

<TEXTAREA ROWS=20 COLS=60 NAME="body">\$body</TEXTAREA><BR>

<INPUT TYPE="submit" VALUE="Send the mail">

<INPUT TYPE="reset" VALUE="Start over"><BR>

<INPUT TYPE="hidden" NAME="nexturl" VALUE="\$nexturl"></P>

</FORM>

<HR>

<H2>Information about the WWW Mail Gateway</H2>

<H3><A HREF="http://www-bprc.mps.ohio-state.edu/mailto/mailto\_info.html#about">

About the WWW Mail Gateway</A></H3>

<H3><A HREF="http://www-bprc.mps.ohio-state.edu/mailto/mailto\_info.html#new">

New in version \$version</A></H3>

<H3><A HREF="http://www-bprc.mps.ohio-state.edu/mailto/mailto\_info.html#misuse">

Please report misuse!</A></H3>

<HR>

<ADDRESS><P><A HREF="/~doug/">Doug Stevenson: doug+\@osu.edu</A>

&lt;/P&gt;&lt;/ADDRESS&gt;

&lt;/BODY&gt;&lt;/HTML&gt;

EOH

;

}

#####

# Method POST implies that they already filled out the form and submitted

# it, and now it is to be processed.

#####

elsif (\$ENV{'REQUEST\_METHOD'} eq 'POST') {

# get all the variables in their respective places

\$destaddr = \$in{'to'};

\$cc = \$in{'cc'};

\$fromaddr = \$in{'from'};

\$fromname = \$in{'name'};

\$replyto = \$in{'from'};

\$sender = \$in{'from'};

\$errorsto = \$in{'from'};

\$subject = \$in{'sub'};

\$body = \$in{'body'};



```

$nexturl = $in{'nexturl'};

$realfrom = $ENV{'REMOTE_HOST'} ? $ENV{'REMOTE_HOST'}: $ENV{'REMOTE_ADDR'};

# check to see if required inputs were filled - error if not

unless ($destaddr && $fromaddr && $body && ($fromaddr =~ /^.+\@.+/)) {

    print <<EOH;

```

Content-type: text/html

Status: 400 Bad Request

```
<HTML><HEAD><TITLE>Mailto error</TITLE></HEAD>
```

```
<BODY><H1>Mailto error</H1>
```

```
<P>One or more of the following necessary pieces of information was missing
from your mail submission:
```

```
<UL>
```

```
<LI><B>To</B>:, the full mail address you wish to send mail to</LI>
```

```
<LI><B>Your Email</B>: your full email address</LI>
```

```
<LI><B>Body</B>: the text you wish to send</LI>
```

```
</UL>
```

```
Please go back and fill in the missing information.</P></BODY></HTML>
```

```
EOH
```

```
exit(0);
```

```

}

# do some quick logging - you may opt to have more/different info written

if ($logging) {

    open(MAILLOG,">>$logfile");

    print MAILLOG "$realfrom\n";

    close(MAILLOG);

}

# Log every CGI variable except for the ones reserved for mail info.

# Valid vars go into @data. Text output goes into $data and gets.

# appended to the end of the mail.

# First, get an ORDERED list of all cgi vars from @in to @keys

for (0 .. $#in) {

    local($key) = split(/=/,$in[$_],2);

    $key =~ s/\+/ /g;

    $key =~ s/%(..)/pack("c",hex($1))/ge;

    push(@keys,$key);

}

# Now weed out the ones we want

@reserved = ('to', 'cc', 'from', 'name', 'sub', 'body', 'nexturl');

local(%mark);

```

```
foreach (@reserved) { $mark{$_} = 1; }
```

```
@data = grep(!$mark{$_}, @keys);
```

```
foreach (@data) {
```

```
    $data .= "$_ -> $in{$_}\n";
```

```
}
```

```
# Convert multiple bodies (separated by \0 according to CGI spec)
```

```
# into one big body
```

```
$body =~ s/\0//;
```

```
# now check to see if some joker changed the HTML to allow other
```

```
# mail addresses besides the ones in %addrs, if applicable
```

```
if (%addrs) {
```

```
    if (!scalar(grep($_." <$addrs{$_}>" eq $destaddr ||
```

```
                $destaddr eq $_, keys(%addrs)))) {
```

```
        print &PrintHeader();
```

```
        print <<EOH;
```

```
<HTML><HEAD><TITLE>WWW Mail Gateway: Mail address not allowed</TITLE></HEAD>
```

```
<BODY>
```

```
<H1>Mail address not allowed</H1>
```

```
<P>The mail address you managed to submit, <B>$destaddr</B>, to this script is
```

```
not one of the pre-defined set of addresses that are allowed. Go back and
```

try again.</P>

</BODY></HTML>

EOH

```
;
```

```
    exit(0);
```

```
}
```

```
}
```

```
# if we just received an alias, then convert that to an address
```

```
$realaddr = $destaddr;
```

```
if ($addrs{$destaddr}) {
```

```
    $realaddr = "$destaddr <$addrs{$destaddr}>";
```

```
}
```

```
# fork over the mail to sendmail and be done with it
```

```
if ($active) {
```

```
    if ($listserver) {
```

```
        open(MAIL, "| $sendmail$fromaddr") ||
```

```
            &InternalError('Could not fork sendmail with -f switch');
```

```
}
```

```
else {
```

```
open(MAIL,"| $sendmail") ||
```

```
&InternalError('Could not fork sendmail with -f switch');
```

```
}
```

```
# only print Cc if we got one
```

```
print MAIL "Cc: $cc\n" if $cc;
```

```
print MAIL <<EOM;
```

From: \$fromname <\$fromaddr>

To: \$realaddr

Reply-To: \$replyto

Errors-To: \$errorsto

Sender: \$sender

Subject: \$subject

X-Mail-Gateway: Doug\'s WWW Mail Gateway \$version

X-Real-Host-From: \$realfrom

\$body

\$data

EOM

```
close(MAIL);
```

```
}
```

```
# give some short confirmation results
```

```
#  
  
# if the cgi var 'nexturl' is given, give out the location, and let  
  
# the browser do the work.  
  
if ($nexturl) {  
  
    print "Location: $nexturl\n\n";  
  
}  
  
# otherwise, give them the standard form.  
  
else {  
  
    print &PrintHeader();  
  
    print <<EOH;
```

```
<HTML><HEAD><TITLE>Mailto results</TITLE></HEAD>
```

```
<BODY><H1>Mailto results</H1>
```

```
<P>Mail sent to <B>$destaddr</B>:<BR><BR></P>
```

```
<PRE>
```

```
<B>Subject</B>: $subject
```

```
<B>From</B>: $fromname &lt;$fromaddr>
```

```
$body</PRE>
```

```
<HR>
```

```
<A HREF="$script_http">Back to the WWW Mailto Gateway</A>
```

```
</BODY></HTML>
```

EOH

;

}

} # end if METHOD=POST

#####

# What the heck are we doing here????

#####

else {

print &lt;&lt;EOH;

&lt;HTML&gt;&lt;HEAD&gt;&lt;TITLE&gt;Mailto Gateway error&lt;/TITLE&gt;&lt;/HEAD&gt;

&lt;BODY&gt;&lt;H1&gt;Mailto Gateway error&lt;/H1&gt;

&lt;P&gt;Somehow your browser generated a non POST/GET request method and it

got here. You should get this fixed!!&lt;/P&gt;&lt;/BODY&gt;&lt;/HTML&gt;

EOH

}

exit(0);

#

# Deal out error messages to the user. Gets passed a string containing

# a description of the error

#

```
sub InternalError {
```

```
    local($errmsg) = @_;
```

```
    print &PrintHeader();
```

```
    print <<EOH;
```

Content-type: text/html

Status: 502 Bad Gateway

```
<HTML><HEAD><TITLE>Mailto Gateway Internal Error</TITLE></HEAD>
```

```
<BODY><H1>Mailto Gateway Internal Error</H1>
```

```
<P>Your mail failed to send for the following reason:<BR><BR>
```

```
<B>$errmsg</B></P></BODY></HTML>
```

```
EOH
```

```
    exit(0);
```

```
}
```

```
##
```

```
## end of mailto.pl
```

```
##
```

If your server can run CGI scripts and is configured with sendmail, this is the right mail gateway script to have in your HTML, you will need to be able to run CGI scripts on your server though.

The use of firewalls can enhance your protection. It can restrict the access of outside mail to only few machines and re-enforce security on those machines. Usually these machines would act as a gateway to the company and a firewall as a guard, a security agent, controlling what's coming in or going out.

Nevertheless, messages will need to come into the company, and a firewall will not be able to screen those messages for hostile applets or scripts. At most, there are few techniques to filter threatening characters in the mail address, if you can come up with a table so that the firewall can recognize it.



Thus, always keep in mind that, since SMTP lacks authentication, forging email is not something difficult. If your site allows connections to the SMTP port, anyone can connect to the that port and issue commands that will send emails that appears to be from you or even a fictitious.

## File Transferring Issues

File transferring is one of the Internet services most used. With the Web, this service became much easier to use, and therefore, more difficult to control and secure. Thus, for security reasons, companies connected to the Internet often block FTP, TELNET, and GOPHER access. Firewalls and proxy servers can protect your site by controlling the access to authenticated FTP sites.

## File Transfer Protocol (FTP)

Security is one of the major opponents of FTP services. Many companies bar FTP fearing been attacked by a hacker, or even having an intruder eavesdropping the site.

Using private FTP over the Internet has some security implications. As with rcp, the user name and password are transmitted in the clear, so anyone on the route between your client and server can sniff your user name and password. They can then use your user name and password to gain unauthorized access to the server. The data you transfer are also unencrypted and can be sniffed as well.

These two problems can be overcome by using a SSL (Secure Socket Layer) version of the FTP server and client program. When using SSL, all network traffic is encrypted, and the client and server can use strong authentication. There is one drawback however, the SSL protocol requires a third, independent party, as a CA (Certification Authority). This CA must be trusted by both parties and is used in establishing the true identity of the client and server. In the case of a Web browser, this CA is one of the "true" authorities, like Verisign is (for more information on VeriSign, check their URL at <http://www.verisign.com>). However, for a dedicated FTP connection between a client and a server this CA can be any party that is trusted by both.

To resolve this problem, there are firewall and proxy products available to incorporate a secured anonymous FTP server, which provides read-only access to a protected and limited file hierarchy. This products provide an interface mechanism that enables a writable incoming directory to allow the sending of files to a firewall. The data areas are then accessed only from the internal network. For more information on firewalls, refer to chapter 14, "Types of Firewalls," where it lists all the main firewall products available on the market.

Try to develop a configuration checklist based on the environment you have, don't go around coping recommendations from books or from the Web! Instead, used them as a template to be customized to the needs and systems characteristics of your company. The following are configuration suggestions to be considered (Remember! Add to the list depending on your needs!):

- **Check if your FTP server is running correctly** - Periodically you should check if your FTP Server service is running correctly. If you are using a Windows NT server, you can try to use FTP on the local system by typing the IP loopback address from the command line:

```
ftp 127.0.0.1.
```

There should be no difference between the interaction with a local server and other Windows NT and most UNIX clients. This can also be used to determine whether the directories, permissions, and so on of the FTP Server service are configured properly.

- **Check if your FTP server is configured right** - If you find any problems after the test above, following CIAC's recommendations, you should consider the following guidelines when configuring your FTP Server:
  - Make sure that files and directories in the anonymous FTP area are not owned by the user "ftp", which is the user ID of anonymous users. The risk is that anything owned by it can be modified, replaced, or deleted by any remote user on the Internet.
  - Make sure not to place any encrypted passwords from the system password file "etc/passwd" password file in the anonymous FTP area "~ftp/etc/passwd." A hacker can retrieve these encrypted passwords and also attempt to decrypt them. Try not to set directories or files as writable for anonymous users. Even though some remote users may find it easier to have an incoming directory available for dropping files, hackers can use these areas to store contraband files, which can include copyrighted materials, etc.
- **Check if your anonymous FTP configuration safe** - Anonymous FTP can be a valuable service at your site, but you must

configure it right and take the time to administer it. Otherwise, you will be opening doors, inviting intruders, hacker and cracker to come in. As a alerted you above, not all the recommendations I am listing here will necessarily apply to you as your environment and/or system platform might differ. So please, remember that:

- Make sure you have the latest version of the FTP daemon/server.
- When setting up FTP directories, make sure the anonymous FTP root directory, "~ftp," and its sub-directories are not owned by the FTP account or even in the same group. Otherwise, as stressed earlier, these can be an open door for attackers, specially if the directory is not write-protected.
- You should have the FTP root directory and its sub-directories owned by root and also have only root with permissions to write on it. This way you will keep your FTP service secure. The following is an example of an anonymous FTP directory structure:

```
drwxr-xr-x  7  root    system  512 Mar  1    15:17 ./
drwxr-xr-x 25  root    system  512 Jan  4    11:30 ../
drwxr-xr-x  2  root    system  512 Dec 20   15:43 bin/
drwxr-xr-x  2  root    system  512 Mar 12   16:23 etc/
drwxr-xr-x 10  root    system  512 Jun  5   10:54 pub/
```

Note that files and libraries, including those used by the FTP daemon and those in ~ftp/bin and ~ftp/etc, should have the same protections as these directories: not be owned by FTP or in the same group and be write-protected.

- Never place system files in the ~ftp/etc directory. It will allow open access to attackers to get a copy of these files. Keep in mind that these files are optional and are not used for access control. Instead, use a dummy version of both the ~ftp/etc/passwd and ~ftp/etc/group files, owned by root. This way, have the dir command using these dummy versions to show the owner and group names of the files and directories.
- Make sure that the ~/ftp/etc/passwd file do not contain any account names already contained in the system's /etc/passwd file. Include on these files only the necessary information to the FTP hierarchy or needed to show owner and group names.
- If you have a firewall set up in place, it is possible for hackers to gain access to your FTP server through the Web, by-passing the firewall. That is one of the reasons why some sites rather have the Web server outside the firewall. Therefore, make sure the password field has been cleared. The following example shows the use of asterisks (\*) to clear the password field. The example was taken from a passwd file at the anonymous FTP area on cert.org:

```
ssphwg:*:3144:20:Site Specific Policy Handbook Working Group::
```

```
cops:*:3271:20:COPS Distribution::
```

```
cert:*:9920:20:CERT::
```

```
tools:*:9921:20:CERT Tools::
```

```
ftp:*:9922:90:Anonymous FTP::
```

```
nist:*:9923:90:NIST Files::
```

It is important to understand that there is a risk in allowing anonymous FTP connections to write to your server. Therefore, you

must evaluate the risks involved before opening the door. Besides the risks already discussed earlier (temporary storage for contraband files, etc.), an attacker could generate a malicious upload of endless files to the point of causing denial of service problems in your server.

## ***Trivial File Transfer Protocol (TFTP)***

But FTP is not the only protocol used to transfer files, as defined in RFCs 783 and 951. Trivial File Transfer Protocol (TFTP) is commonly used by dedicated devices to transfer configuration files.

If you are running TFTP on a UNIX system, turn it off! TFTP provides significant security risks. If you take the AIX version 3.x, for example, it allows remote users to upload /etc/passwd.

Also, there are scanners, such as NSS (Network Security Scanner) and CONNECT, that will specifically search for open TFTP holes. If you must run TFTP, make sure to,

- Use shadowed password,
- Run TFTP in a secure mode by setting it in inet.conf with the -s option,
- Log and check your connections daily!

### **Tip:**

You can download Joe Hentzel's TFTP CONNECT scanner from the URL <http://www.giga.or.at/pub/hacker/unix/>.

## ***File Service Protocol (FSP)***

File Service Protocol (FSP) is very similar to FTP in the way or works and its features. However, FSP has protection against network overload (never forks) and logs the username of the connection coming in to the server. There is a scanner, called FSPScan, developed by Wen-King Su, that scans for FSP servers. You can download it from the URL <http://www.giga.or.at/pub/hacker/unix>.

## ***UNIX-to-UNIX Copy Protocol (UUCP)***

UNIX-to-UNIX Copy Protocol (UUCP) is a software program that facilitates file transfer from one UNIX system to another UNIX system via dial-up phone lines. UUCP protocol also describes the international network used to transfer USENET News and electronic mail.

If using UUCP, make sure to disallow name service, as you don't want to be giving out potentially\_ compromising information. In general, you don't want people to know what the internal structure of your network really is. Also, for any open port above 1023, as long as your system isn't listening on a port, that port is not vulnerable.

Nonetheless, try to use a proxy server, rather than allowing the packet through directly. This allows some logging, possibly some action to be taken on the firewall.

## ***The Network News Transfer Protocol (NNTP)***

NNTP is a protocol used for moving around Usenet News, a bulletin board-like on the Internet, with a variety of articles in many subjects. Grouped into newsgroups, the articles are selected by their content.

When setting up news to be accessed through your Web server, you will use NNTP to link news to your site. You will have to decide where your news server will be located in order to preserve security. Assuming that you have or will be installing a firewall at your site, you have the option to place NNTP at the firewall machine, the bastion host. Or you can have NNTP outside of your protected network, if your Web server is placed outside of it.

However, to secure news links is not something difficult to do. The major issue you will face is on controlling the private news

your internal users may create. Chances are your users will be exchanging sensitive information among each other, but if external users have access to these groups, then you may have a breach of confidentiality situation to deal with. NNTP can help you to control access to these private groups.

The proxying capabilities of NNTP can help you to filter the Usenet news postings by receiving and storing it than forwarding it to a server you have designated.

NNTP is a TCP-based service with store-and-forward characteristics protocol. For the most part, NNTP is a very secure protocol, carrying a very secure service. The reason being is that all the incoming connections to your site will be coming from a licit connection from a news feed location.

Regardless where you place your firewall, make sure to have the news being feed straight from your news provider to your news server. You will be able to do this very easily by using packet filtering or, in case you have a firewall, through a proxy server.

Although NNTP is a fairly secure protocol and easy installation, the following are few recommendations you should keep in mind when configuring news at your Web site:

- Do not allow the news server to reside in your firewall (bastion host) machine. News are very disk storage demanding, and you don't want to have your firewall machine crashing do to lack of disk space!
- Make sure to disable automated group creation. It can present a risk to your site as if groups are not created properly it could enable (specially in UNIX) commands to be issued.

If you ever decide to install Usenet-Web, make sure not to run the `usenet-web-index-rebuild.pl` program at the same time as the `usenet-web-archiver.pl`.

Also, make sure to disable any cron jobs that could be running the `usenet-web-archiver.pl` before you run `usenet-web-index-rebuild.pl`.

It might look obvious to place the news server on your firewall machine, as discussed above, avoid doing so. If you must, then you may want to consider a dual firewall system, which will increase cost and maintenance.

If you are using a firewall at your site, one of the easiest way to configure your news gateway is through packet filtering. The following is a small list of recommendations:

- The packet filtering should allow incoming NNTP connections carrying news from ports above 1023 or the remote system and 119 on your Web or news server. Conversely, you should set packets with the ACK bit set from port 119 on your Web server to ports equal or greater then 1023.
- By the same token, make sure your NNTP outgoing connections are set from packets 1023 on your news server to the remote system, on port 119. As above, the packets with ACK bit set from 119 on the remote system to port above 1023 on your Web server.
- Make sure your news gateway is compatible with your firewall. Remember that I the HTTP environment, clients use ports above 1023, which without the gateway could be problem.

## The Web and the HTTP Protocol

The Hypertext Transfer Protocol (HTTP) is an application-level protocol developed for distributed, collaborative, hypermedia information systems. The HTTP protocol is very generic and stateless, allowing systems to be built independently of the data being transmitted. It is also an object-oriented protocol with capabilities to be used for a variety of tasks, which includes but is not limited to name servers, distributed object management systems and extension of its request methods, or commands.

One of the great features of HTTP is the typing and negotiation of data representation. This protocol has been in use since 1990, with the W3 global information initiative.

The most current version of HTTP is version 1.0, which is supported by all Web servers in the market. But there is also another version of the protocol, HTTP-NG (Next Generation), which promises to use the bandwidth available more efficiently and enhance the HTTP protocol.

Further, HTTP is a protocol that can be generically used for communication between user agents and proxies or gateways to other Internet protocols, such as SMTP, NNTP, FTP, Gopher and WAIS.

Nevertheless, all this flexibility offered by HTTP comes at a price: it makes Web server, and clients, very difficult to secure. The

openness and stateless, characteristics of the Web, accounts for its quick success, but makes it very difficult to control and protect.

On the Internet, HTTP communication generally takes place over TCP/IP connections. It uses as default port 80, but other ports can be used, which does not prevent HTTP from being implemented on top of any other protocol. In fact, HTTP can use any reliable transport.

When a browser receives a data type it does not understand, it relies on additional applications to translate it to a form it can understand. These applications are usually called viewers, and should be the one of the first concerns you should have when preserving security. You must be careful when installing one, because, again, the underlying HTTP protocol running on your server will not stop the viewer from executing dangerous commands.

You should be especially careful with proxy and gateway applications. You must be cautious when forwarding requests that are received in a format different than the one HTTP understands. It must take into consideration the HTTP version in use, as the protocol version indicates the protocol capability of the sender. A proxy or gateway should never send a message with a version indicator greater than its native version. Otherwise, if a higher version request is received, both the proxy or the gateway must either downgrade the request version, respond with an error, or switch to a tunnel behavior.

**Tip:**

If you need more information on HTTP, check the URL:

<http://www.w3.org/hypertext/WWW/Protocols/>

There is a series of utilities intended for Web server administrators available at the URL:

<ftp://src.brunel.ac.uk/WWW/managers/>

## ***Proxying HTTP***

The majority of HTTP clients, such as Purveyor and Netscape Navigator, support a variety of proxying schemes, SOCKS and transparent proxying.

Purveyor, for instance, provides proxy support for not only HTTP, but also FTP and GOPHER protocols, creating a secure LAN environment by restricting Internet activities of LAN users. The proxy server offers improved performance by allowing internal proxy caching. Purveyor also provides proxy-to-proxy support for corporations with multiple proxy servers.

**Tip:**

For more information on Purveyor Webserver, check Process Software's URL:

<http://www.process.com>.

If you are running your Web server on Windows NT, Windows 95 or NetWare, you can use Purveyor Webserver's proxy features to enhance security. In addition, you can increase the performance of your server as Purveyor can locally cache Web pages obtained from the Internet.

You should consider installing a firewall at your site, regardless if you are placing your server outside or inside your protected network. The openness of HTTP is too great for you to risk. Besides, you still have all the viewers and applets to worry about.

When selecting a firewall, make sure to choose one that includes HTTP proxy server. It will be useful for protecting your browsers. Some firewalls, such as the TIS Firewall Toolkit, provide HTTP proxying totally transparent to the user.

## ***HTTP Security Holes***

The HTTP protocol has some more security holes to justify a firewall. One of them is that it allows remote users to request communication to a remote server machine, and to execute commands remotely. This security hole compromises the Web server and the client in many ways, including but not being limited to:

- Arbitrary authentication of remote requests.

- Arbitrary authentication of Web servers.
- Breach of privacy of request and responses.
- Abuse of server features and resources.
- Abuse of servers by exploiting its bugs and security holes.
- Abuse of log information (extraction of IP addresses, domain names, file names, etc.)

Most of these security holes are well known. Some applications like Netscape's SSL and NCSA's S-HTTP try to address the issue, but only partially.

Web servers are very vulnerable to client's behavior over the Internet. Therefore, clients should prompt a user before allowing HTTP access to reserved ports other than the port reserved for it. Otherwise, these could cause the user to unadvertedly cause a transaction to occur in a different and danger protocol.

You must be careful also with the GET and HEAD methods! The so trivial link to click an anchor to subscribe or reply to a service can trigger an applet to run without the user's knowledge, which enables the abuse by malicious users.

Another security hole of HTTP has to do with server logs. Usually, a Web server logs a large amount of personal data about information requested by different users. Evidently, this information should remain confidential. HTTP allows the information to be retrieved without any access permission scheme.

Many other HTTP limitations and security holes exist if we were to break down the ramifications of the above security issues presented by the protocol. Here are few HTTP configuration checklist to help you out:

- When configuring your HTTP server, never use raw IP addresses to allow access to your pages. Otherwise, you will end up with a bunch of them in your access list, which only will make maintenance harder.
- If you ever have problems with misconfigured clinet's domain server, have them contacting the LAN or systems administrator to fix it so you can reverse their names correctly. If you are the one to fix the problem, take the time and do it! In a the long run you will be thankful for it as otherwise, you may end up with a huge list of raw IP addresses on your list.
- You have to deal with access.conf files, make sure to put only one name per directive, which will ease the file editing, as you can comment out any directive by simply placing the "#" character at the start of the line.
- Remember to reboot your server after any changes made on your access.conf, as the changes you made will not take effect until you re-start the system.
- Always have an access control list of the top-level document directory. It will be useful when updating the file later.

## Security of Conferencing

Of course, there must be a practical reason for you to use the Web for conferencing. Not only there is a large variety of hardware and software, but the fact that the Web provides a common user interface for Internet utilities like FTP, Telnet, Gopher, and WAIS allows the users to reach all the resources available on the Internet without having to leave the Web.

Despite the advances of Web technology in the past three or four years, there are still a series of issues to be addressed before considering conferencing, at least in a large scale. The following is a summary list of the main challenges affecting Web conferencing deployment:

- **Freshness of information** - Just like with news gateways, users want to read only the new messages added since their last visit. In the Web environment, either the client or the server could do this.
- **Ability to submit files to the system** - Users should be able to upload files onto the conferencing system. To have to type it all over again in the Web form is unproductive.
- **Incorporate images and sound to the messages** - Thus one of the most exciting features enabled by the Web, image and sound is one of the most difficult to implement. As long as an image is already available on a Web server, you can link any HTML message to it.
- **Risks of HTML usage** - It might seem natural to allow users to manipulate HTML markups in their messages, but it may create a formatting problem, as users may produce messages not compatible with your conferencing application. Users would have to be aware of structural elements such as message headers and navigation buttons.
- **Keeping users on track** - On the Web, it is very easy for an user to take side trips by clicking on a hyperlink. This could be a problem if these links were to be appended to the message.
- **Speed** - To carry sound and image on the Web can be a problem for users with low bandwidth connection. A 14.400 baud

modem on the Web can be awfully slow when transferring image and sound.

The bottom line, you must take in consideration the clientele accessing your site, the Web conferencing technology to be deployed and the bandwidth you have available to deploy this service. Conferencing involves skimming over a lot of stuff to find the most interesting nuggets, so you need to be able to move around quickly.

## Watch This Services

Besides all, you should keep an eye on these following services, as they also can affect the security of your site if you don't configure them appropriately

### *Gopher*

Gopher is not as used as before, but it is still fast and efficient. Believe it or not, Gopher is fairly secure but there are some issues I would like to alert you about. One of the most popular Gopher server is the one of the University of Minnesota (found at [boombox.micro.umn.edu](http://boombox.micro.umn.edu)), which is run by a lot of the Gophers available out there.

You should know that there is a bug on both Gopher and Gopher+ in all versions that were available before August of 1993, as reported in CERT Advisory CA-93:11. This bug allows hacker to obtain password files, both remotely or locally, by potentially gaining unrestricted access to the account running the public access client and reading any file accessible to this account. This includes the `/etc/passwd` and other sensitive files.

If you want to review this bug, you can check it at the Defense Data Network Bulletin 9315, which can be viewed at the URL [HTTP://www.arc.com/database/security\\_bulletins/DDN/sec-9315.txt](http://www.arc.com/database/security_bulletins/DDN/sec-9315.txt).

You should be alert about Gophers proxying an FTP session. Even if access is restricted to an FTP directory on your server, the Gopher can be used to perform a bounce attack. Thus, be careful when protecting an FTP server behind a firewall. If the Gopher server is not protected, a hacker can use it to trespass the firewall.

Another vulnerability, reported by NASA Automated Systems Incident Response Capability (NASIRC), indicates a failure in the gopher servers `gpopher1.1` (Gopher) and `gopher2.012` (Gopher+) internal access controls, which can allow files in directories above the gopher data directory, such as the password file, to be read if the gopherd does not run `chroot`. This vulnerability only affects servers that are started with the option `"-c"`. Without this option, gopherd runs `chroot` and access to files above the gopher-data directory is disabled.

### *finger*

Finger is a program that tells you whether someone is logged on to a particular local or remote computer. Through finger you might be able to learn the full name, terminal location, last time logged in, and other information about an user logged onto a particular host, depending on the data that is maintained about users on that computer. Finger originated as part of BSD UNIX.

To finger another Internet user, you need to have the finger program on your computer or you can go to a finger gateway on the Web and enter the name of the user. The user's computer must be set up to handle finger requests. A `".plan"` file can be created for any user that can be fingered.

An intruder can use finger to find information about a site, and use finger gateways to protect his identity.

### *whois*

Whois is a program run by InterNIC that will tell you the owner of any second-level domain name. For example, you can look up the name of the owner of your own access provider by entering for example, `"process.com"` and whois will tell you the owner of that second-level domain name. The InterNIC Web whois is at <http://rs.internic.net/cgi-bin/whois>.

whois can also be used to find out whether a domain name is available or has already been taken. If you enter a domain name you are considering and the search result is `"No match,"` the domain name is likely to be available and you can apply to register it through your service provider.

The security risk with whois is that a hacker can look-up information about his/her target before striking. As a matter of fact, this information can be used for exploring security weaknesses in your system.

For instance, there is program on a Gopher server that will produce similar results as whois, but this one will tell you the names of all domain name holders associated with a specific second-level domain name. This program is at `gopher://rs.internic.net/7waissrc%3A/rs/whois.src`. At IBM, for example, you can lookup information about its employees by checking their whois service at `http://whois.ibm.com`. The same goes for Stanford University, which you can look-up information about their students.

## ***talk***

Talk is a UNIX service that allows two users to communicate over the Internet via text-based terminals. It's much similar to the Net send command and IRC, only that the connection is directed by the persons e-mail address. Thus, if you were to talk to me via the Internet you would issue a command:

```
talk goncalves@process.com
```

By issuing this command the local talk program would contact the remote talk daemon. If I'm available, assuming that I have talk connections enabled, my screen would split and conversation would take place. If you're familiar with the chat command of Windows for Workgroups, bundled with the network tools, you know what I'm talking about.

The risk with this service is that information can be gathered from an unadvertised user that engage in conversation with someone unknown out on the Internet.

## ***IRC***

Internet Relay Chat (IRC), just like talk, allows the communication over the Internet. However, IRC allows multiple users conversing at the same time.

The main risk is that file transferring can be done over IRC without any traced left behind, its like a cash transaction without receipts! Even though these file transferring can be done through FTP, etc., IRC makes it possible without any server software running.

## ***DNS***

As you already know, Domain Name System (DNS) is the way that Internet domain names are located and translated into Internet Protocol (IP) addresses. Because maintaining a central list of domain name/IP address correspondences would be impractical, the lists of domain names and IP addresses are distributed throughout the Internet in a hierarchy of authority. There is probably a DNS server within close geographic proximity to your access provider that maps the domain names in your Internet requests or forwards them to other servers in the Internet.

As far as risks with DNS, you should be aware of spoofing. When a DNS machine is compromised, this machine has been a victim of a spoofing. Not that it happens very often, but there has been reports, both at DDN and CIAC, about DNS spoofing.

CIAC's advisory, entitled "Domain Name Server Vulnerability alerts about the possibility of an intruder to spoof BIND into providing incorrect name data at the DNS server, allowing for unauthorized access or re-routing of connections. Can you imagine if all private connections of the Secret Services were re-routed to a hackers home server? Fortunately (or should I say hopefully), the Secret Service is already using Skipjack or some other kind of strong encryption in their IP connections!

But fear not! A DNS spoofing is not an easy task. It's not enough for an intruder to gain access to the DNS server. The intruder will have to re-route the addresses of that database, which would easily give him away. It's like breaking the window of a jewelry store, it's just a matter of minutes before the police arrives. But again, with a good plan, how much time would a hacker need to get what he wants?



# Network Management Station (NMS)

As described by Aday Pabrai and Vijay Gurbani in their book "Internet and TCP/IP Network Security, by McGraw-Hill, "Network Management Station (NMS) is a system responsible for supporting a network management protocol and applications necessary for it to process and access information from entities (managed nodes) on the network."

The only security feature provided by NMS is access control. NMS, additionally provides authentication and privacy.

## Simple Network Management Protocol (SNMP)

Simple Network Management Protocol is the protocol governing network management and the monitoring of network devices and their functions. It is not necessarily limited to TCP/IP networks. The details of SNMP are in these Internet Engineering Task Force (IETF) Requests for Comments (RFCs):

There are two versions of SNMP, SNMPv1 and SNMPv2. SNMPv1 is the older of the two SNMP versions, of course, and offers very rudimentary security features. The only security feature offered by SNMPv1 is that of access control. In an SNMPv1 environment there are a number of agents that are monitored or controlled by a manager. Thus a manager contains a set of agents.

At this stage two concepts can be introduced firstly, an MIB should be viewed as a database with tables and relationship between the tables. Secondly the concept of community. An SNMP community is a relationship between an SNMP agent and a set of SNMP managers that defines authentication, access control, and proxy characteristics. The community is established locally at an agent and is given a name. The community is addressed by its name. Thus a community is a relationship between an agent and a manager for certain privileges of the agent MIB.

### Tip:

What is a MIB? A Management Information Base (MIB) is a formal description of a set of network objects that can be managed using the Simple Network Management Protocol (SNMP). The format of the MIB is defined as part of the SNMP. All other MIBs are extensions of this basic MIB. MIB-I refers to the initial MIB definition. MIB-II is the current definition. SNMPv2 includes MIB-II and adds some new objects.

There are MIB extensions for each set of related network entities that can be managed. For example, there are MIB definitions in the form of Requests for Comments (RFCs) for Appletalk, DNS server, FDDI, and RS-232C network objects. Product developers can create and register new MIB extensions.

Companies that have created MIB extensions for their sets of products include Cisco, Fore, IBM, Novell, QMS, and Onramp. New MIB extension numbers can be requested by contacting the Internet Assigned Numbers Authority (IANA) at 310-822-1511 x239.

The SNMPv2 Working Group recently completed work on a set of documents which makes up version 2 of the Internet Standard Management Framework. Unfortunately, this work ended without reaching consensus on several important areas -- the administrative and security framework and remote configuration being two of the most important.

The IETF has charted a Working Group to define SNMPv3, which, if successful, will replace the SNMPv2. The SNMPv3 effort has been underway since April 1997.

## traceroute

Van Jacobson is the author of traceroute, which is a tool to trace the route IP packets take from the current system to some destination system. What it does is, by using the IP protocol "time\_to\_live" field it attempts to elicit an ICMP TIME\_EXCEEDED response from each gateway the packet goes through on its way.

The danger here is that this utility can be used to identify the location of a machine. Worse, you don't even need to run Unix to have access to traceroute. There are several gateways on the Net, such as the one at the URL <http://www.beach.net/traceroute.html>.

Figure 8.01 is a traceroute to my server at Process Software Corp.

## Network File System (NFS)

Network File System (NFS), was popularized by Sun to provide a shared file system for UNIX machines. NFS, like its relative NIS, is based on a trust model of network machines that exchange information based on account information. NFS only allows certain machines to access shared file systems, but determining which machines are allowed to access the file systems is accomplished by a simple lookup of the address of the accessing machine, which can be done by anyone with access to the system running NFS.

A system can be impersonated by another system to obtain its rights to a file system. This was one of the strategies used by Kevin Mitnick to break into systems, and how NFS systems are commonly attacked.

If you are to use NFS, employ NFS version 3, which can handle encryption and much stronger authentication of connecting machines. Distributed file systems are historically vulnerable, but as a UNIX standard and as widely deployed as it is in educational and research arenas, NFS tends to gain more than its fair share of examination and dissection.

NFS is one of the most important and vulnerable network service in Sun's system, as it provides full access to files and directories. The major security hole is that NFS's access control mechanisms are very hard to maintain, and are hardly adequate. Another hole is that it doesn't have user authentication, even when using the so-called secure NFS implementation.

Every user can write his own NFS client, specify any identity and read or write files. An NFS client that provides this basic functionality can easily be written in about 300 lines of C code. The secure NFS tries to fix this security hole but it doesn't totally succeeds. The problem is that the underlying cryptosystem doesn't work, and can be broken very easily.

File handles also used to (it has been fixed!) represents a major vulnerability. They can be constructed without the help of the mount daemon, which allows a client to directly go to the NFS daemon, and bypass the access control mechanisms which are enforced by the mount daemon.

Nowadays, hackers are very aware of the typical security models utilized by MIS and deployed all over the Internet. Hackers can write simple applets to act as NFS clients and bypass all the access control system normally used, gaining total access to internal networks or users files. But this is not merely a security hole of NFS, it extends to almost every network service available.

## Confidentiality and Integrity

The Internet itself will not protect your confidential or sensitive information. If you don't take care of it, nobody will! The fact that neither users or Internet providers are regulated makes security even more difficult, because the Internet is open to everyone. It is like trying to protect your home, but without any locks on the doors.

Authentication mechanisms are very important to safeguard the integrity, confidentiality and security of your users, specially if you are involved with electronic commerce, which becomes a requirement. Therefore, clients must authenticate themselves to Web servers, and Web server must also authenticate themselves to clients, and that both authenticate to each other. When applying authentication methods, it is important to take into consideration the spoofing risks. Cryptography methods, as discussed on chapter 3, "Cryptography: Is It Enough?," will help you to implement a security policy not so easy to be spoofed by hacker.

Confidentiality, is also very important for users dealing with sensitive data. Again, the credit card example comes to mind, your account number would be the last thing you want publicized! In the corporate world, this requirement will be amplified as financial data, marketing and sales forecasts are exchanged over the Web. The data traversing the Web needs to be protected.

### Note:

Enterprise Integration Technologies, Inc. (EIT) is designing a Secure HTTP, a set of protocol changes to address confidentiality, integrity, and authentication issues. For more information you can check their Web page at URL: <http://www.eit.com>

As for integrity, just keep in mind that certain transactions not only require confidentiality, but also that contents will not be modified. The banking industry, for example, relies on confidentiality, but the integrity of the data is as important as the privacy of

the information being exchanged.

There are tools to help you preserve the confidentiality and integrity of your connections. Firewalls and encryption are definitely necessary, but you can also increment security by using tool such as swIPe, developed by John Ioannidis. This tool is actually a a network-layer security protocol for the IP protocol suite. swIPe provides confidentiality, integrity, and authentication of network traffic and can be used to provide both end-to-end and intermediate-hop security. swIPe is concerned only with security mechanisms; policy and key management are handled outside the protocol.

SwIPe is a network level encryptor of datagrams, not a simple application level process. Be advised that the secure use of swIPe also requires other problems to be solved, such as key management, which are far beyond what many firewalls are instructed to do. You could be a little bit creative and try to splice in user level encryption into the firewall, but this would not be swIPe. It would also increase their complexity somewhat, decreasing confidence in the security of the modules themselves.

If you want to use swPIe with a firewall, I know for a fact that Gauntlet Internet Firewall runs on BSD/OS and uses swIPe.

Orders

Orders

Back

Forward

Comments

Comments

---

[COMPUTING MCGRAW-HILL](#) | [Beta Books](#) | [Contact Us](#) | [Order Information](#) | [Online Catalog](#)

---

HTML conversions by [Mega Space](#).

This page updated on December 05, 1997 by [Webmaster](#).

[Computing McGraw-Hill](#) is an imprint of the [McGraw-Hill Professional Book Group](#).

Copyright ©1997 [The McGraw-Hill Companies, Inc.](#) All Rights Reserved.

Any use is subject to the rules stated in the [Terms of Use](#).

|        |        |
|--------|--------|
| Orders | Orders |
|--------|--------|

|      |         |
|------|---------|
| Back | Forward |
|------|---------|

|          |          |
|----------|----------|
| Comments | Comments |
|----------|----------|

© 1997 The McGraw-Hill Companies, Inc. All rights reserved.  
Any use of this Beta Book is subject to the rules stated in the [Terms of Use](#).

---

## Chapter 9

# Setting Up a Firewall Security Policy

To talk about security policy we must talk about risks. As it is the antithesis of security, we naturally strive to eliminate risk. As worthy as that goal is, however, we learn with each experience that complete elimination is never possible. Even if it were possible to eliminate all risk, the cost of achieving that total risk avoidance would have to be compared against the cost of the possible losses resulting from having accepted rather than having eliminating risk. The results of such an analysis could include pragmatic decisions as to whether achieving risk avoidance at such cost was reasonable. Applying reason in choosing how much risk we can accept and, hence, how much security we can afford is risk management.

Did you ever heard about "security through obscurity"? Although it is not as evident within many organizations (it is obscure!), this security practice used to be very common, and it is still around.

Security through obscurity defines a security system that promotes security by isolating information about the system it is protecting from anyone outside the implementation team. This includes but is not limited to hidden passwords in binary files or scripts in the assumption that no one will ever find it.

Are you running your internal network and are planning to run your firewall based on such a system? Better not! It certainly worked back there with proprietary and centralized systems, back in the "glass walls" age. But today, with the advent of open systems, internetworking, and great development of intelligent applications and applets, security policies needs to be taken a step higher.

To run your site based on hidden information, rather than protected information, is to play with fire (without the wall!). Nowadays, users are more knowledgeable about the systems they are running and the technology surrounding them. To keep information unknown is just a matter of time until it becomes well known. To base security on it is useless.

Hackers are proud to prove that. They were the first ones to prove that obscurity, rather than security, is exciting. Consequently, you will need a system that is genuinely secure. True, it can still be broken, but by been structured you will be dealing with a organized method, where you can have tools to increase security, monitor threats, catch intruders, or even pursuit them.

You must keep your firewall (and protected network!) logically secure. Logic should be your starting

criteria in putting together a security policy that will use algorithmically secure systems such as Kerberos, PGP and many others.

All right, you already know that your site must be secured and also know what needs to be protected. But what makes a site insecure in the first place? Truly? It is the fact that you turned it on!

Your site will be as secure as the people you allow, or invite, to access it. You can have a very secure site where only corporate users have access to it and you have enough information about each one of them (should you need to track them down later!). So let's assess your corporate security...

### ***(c ) Assessing Your Corporate Security Risks***

It is useful in thinking about risk management to use a sort of formula. This is not, of course, a mathematical equation for use in making quantitative determinations of risk level. It is an algorithm for use in thinking about the factors that enter into risk management and in assessing the qualitative level of danger posed in a given situation.

Reliability and the steps necessary to allow for and deal with reliability failures are risk management issues you must take into consideration. In information systems security, the word "threat" describes a more limited component of risk. For these purposes, threats are posed by organizations or individuals who both intend us harm and have the capability to accomplish their intentions.

In order to develop a thorough security policy, you must consider the possible consequences of attacks from a wide variety of different threats, each of which may act on a specific vulnerability different from those attempted to be exploited by other independent threats and any of which may be unrecognized. Often threats to information and information systems are paired with a specific line of attack or set of vulnerabilities - since a threat which has no vulnerability it is capable of exploiting creates no risk, it is useful to deal with threat-vulnerability pairings in the risk management process.

This uncertainty is a contributing cause of our tendency to rely on risk avoidance. By assuming the threat to be capable, intent, and competent, by valuing our potential targets highly, and by conservatively estimating uncertainties, we reduce risk management to: "what are our vulnerabilities and how much do countermeasures cost to eliminate them?" The management problem is, "How much money can I spend and where can I spend it most wisely?" In most cases, fortunately, it is possible to do better. It is often sufficient to bound the problem, even when exact figures are not available. By careful analysis, we may be able estimate the value of each factor in our equation and balance the risk of loss or damage against the costs of countermeasures and select a mix that provides adequate protection without excessive cost.

Ultimately, the risk management process is about making decisions. The impact of a successful attack and the level of risk that is acceptable in any given situation are fundamentally policy decisions. The threat is whatever it is and while it may be abated, controlled or subdued by appropriate countermeasures, it is beyond the direct control of the security process. The process must focus, accordingly, on vulnerabilities and countermeasures. Vulnerabilities are design issues and must be addressed during the design, development, fabrication and implementation of our facilities, equipment, systems and networks. Although the distinction is not always certain, countermeasures are less characteristics of our systems than of their environments and the ways in which we use them. Typically, to make any asset less vulnerable raises its cost, not just in the design and development phase but also due to more extensive validation and testing to ensure the functionality and utility of security features, and in the application of countermeasures during the operation and maintenance phase as well.

Your basic security requirement should be to minimize, if not eliminate, all the security holes existent in your site. This security holes usually are presented in four ways:

1. **Physical** - Caused by unauthorized people accessing the site, enabling them to peruse where they are not supposed to. A good example of this would be a browser placed into a public place (reception area, for example), giving an user the chance not only to browse the Web, but also change the browser's configuration, get site's information such as IP addresses, DNS entries, etc.
2. **Software** - Caused by "buggy privileged" applications such as daemons, for example, executing functions they were not supposed to. As a rule of thumb, never trust scripts and applets! When using them make sure you understand what they are supposed to do (and what they are not!).
3. **Incompatibility Issues** - Caused by bad system's integration planning. A hardware or software may work great alone but once you put it together with other devices, as a system, it may present you problems. This kind of problems are very hard to spot once the parts are integrated into the system. So make sure to test every component before integrating it into your system.
4. **Lack of a security policy** - It does not matter how secure your password authentication mechanism is if your users use their kids names as their passwords. You must have a security policy addressing all the security requirements for your site as well as covering, and preventing, all the possible security roles.

The requirements to run a secure firewall also includes a series of "good habits" that you, as administrator should cultivate. It is a good policy to try to keep you strategies simple. It is easier to maintain, as well as to be modified, if necessary.

Most bastion hosts and firewall applications, as mentioned earlier, have the capability to generate traffic logs. Users are at the mercy of these servers, especially Web servers, when information about themselves, their connections, their address or even specifications about their client or company are disclosed. The log provided by a Web server can be threatening for an user as it disclose a list of information, which usually includes:

- The IP address,
- The server/host name,
- The time of the download,
- The user's name (if known by user authentication or, with UNIX, obtained by the identd protocol),
- The URL requested,
- The data variables submitted through forms users usually fill out during their session,
- The status of the request, and
- The size of the data transmitted.

A fundamental problem of developing a security policy then, is to link the choice of design characteristics which reduce vulnerabilities and of countermeasures to threat and impact in order to create a cost-effective balance which achieves an acceptable level of risk. Such a process might work as follows:

1. **Assess the impact of loss of or damage to the potential target.** While the impact of the loss of a family member as a parent is beyond measure, the economic value of the member as a wage earner can be estimated as part of the process of deciding the amount of life insurance to purchase,

- Table 9.1 provides a matrix to assess the level of security you may need to implement, based on the level of concern with the information to be protected and the potential consequences in case of breach of confidentiality.

**Level of Concern / Suggested authentication method**

|                                                                           |                                                                                                                                                                                                                                                                                                                                             |
|---------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| High-Classified / Use of encryption methods along with packet filtering   | <p>If loss of integrity at your site will affect confidentiality, then the requirements for integrity is high and must be met.</p> <p>If loss of integrity at your site does not affect confidentiality, then your site can be accommodated in one of the requirements below for Low, Medium, or High levels of concern, as applicable.</p> |
| High / Use of encryption methods and associated authentication methods    | Absolute accuracy required for mission accomplishment (e.g. electronic commerce); or expected dollar value of loss of integrity is high.                                                                                                                                                                                                    |
| Medium / Use of authentication methods                                    | High degree of accuracy required for mission accomplishment (personal information being cataloged, health environments), but not absolute or expected dollar value of loss of integrity is not high.                                                                                                                                        |
| Low / Use of password protection                                          | Reasonable degree of accuracy required for mission accomplishment (database applications, search engines); or expected dollar value of loss is low.                                                                                                                                                                                         |
| Very Low / may not require security measures other than integrity of data | No particular degree of accuracy required for mission accomplishment (informative pages, minimum interaction with user).                                                                                                                                                                                                                    |

**Table 9.1**

*Level of Integrity To Be Implemented*

## **Data Security**

Remember! Bastion hosts and servers alike for that matter, are dull! They are obedient, and will do what you ask them to do, but unfortunately, they are dull! Since they will not think on they own, they don't know the difference between the firewall administrator and a hacker (well, we probably wouldn't know either!). Anything placed into the bastion host's document root directory is exposed and unprotected if you don't find a way to protect it.

Bastion hosts that are loaded with a whole bunch of optional features and services are specially prone to data security risks, especially if your bastion host is also your Web server! Many of the features of a Web server that adds convenience and user-friendliness are more susceptible to security flaws. Unfortunately, most of the Web servers software available in the market do not provide any kind of proxy support.

However, according to a multi-platform Web servers comparative review article, written by Jim Rapoza,



for PCWEEK (April 1, 1996), from the six Web servers he reviewed, only two had proxy support. They were IBM's Internet Connection Secure Server for OS/2 Warp 1.1 and Process Software Corp.'s Purveyor Webserver for NetWare 1.0, where Purveyor was rated as having an "excellent" proxy support and Internet Connection a "good" one.

Both products have a very easy interface for setting up proxies, which should be a must feature for your Web site. These products even allow you to cache specific Uniform Resource Locators (URL) and redirect them to other proxy servers, which is a great security feature. Purveyor even allows you to block an internal user to access non-business related Web sites or controversial ones. I am sure you don't want upper management blaming you for allowing employees to spend chunks of the time they should be working accessing Playboy, Penthouse or neonazism sites!

Nevertheless, when choosing a Web server software, have data security in mind, and site security as well! Make sure they have solid access security options. You should be able to set users access parameters and block access to the site based on the IP address or domain name of the client.

Proxy support will help you to prevent attacks or unwanted visitors, enhancing data security. It also will help you to cope with the holes generally opened by dangerous features present in so many Web server software. Another important aspect to consider is the underlying operation system. The operating system underlying the Web server is a vital aspect in determining how safe the server is against hackers attack.

The inherent openness of a UNIX systems, for example, will bring extra work for you when trying to block access to hackers. Conversely, a Mac-based system is a much more secure system as it does not as open as UNIX. Servers running Windows NT, or even Windows 95 or Novell have a good built-in security. One of Windows NT advantage is that it supports a large variety of Web server software, which allows you to tailor your server's configuration.

Besides the operating system, you should be careful with the features each operating system, combined with Web servers had to offer. There are potentially dangerous ones that you should turn off, specially if you do not need them. The following is a list of features that you should pay special attention:

- **Automatic directory listings** - The more a hacker knows about your system the more chance for you to be tampered. Of course, automatic directory listing can be very convenient, but hackers can have access to sensitive information through them. For example:
- Emacs backup files containing CGI scripts,
- Control logs,
- Directories with temporary files, etc.

Be aware that by turning off automatic directory listings won't stop hackers from grabbing files whose names they guess at, but it at least make the process more difficult.

- **Symbolic links following** - There are servers that allow you to extend the document tree with symbolic links. Although convenient, it can become dangerous if the link is created to a sensitive area such as /etc.
- **Server side includes** - One of the major security hole is the "exec" form of server side includes. It should be turned off completely or made available only to trusted users. Apache and NCSA allows you to turn it off by entering the statement in the directory control section of access.conf: Options IncludesNoExec

- **User-maintained directories** - Although it might be useful to allow users to maintain directories at the server, or even to add documents to the Web site, you must trust the users. It would be useful to have all the their publishing files and CGI scripts directly maintained by their authors, but it could create major security holes. You will be better off to give the authors their own space and move them when necessary.

Another way to protect data is through the use of SSL, which uses a public-key encryption to exchange a session key --used to encrypt the http transaction-- between the client and server. Since each transaction uses a different session key, even if a hacker decrypts the transaction, the server's secret key will still be protected.

Netscape servers and browsers do encryption using either a 40-bit secret key or a 128-bit secret key. However, most Netscape users have browsers that support only 40-bit secret keys, due to government restrictions about software that can be exported, but this policy had been modified since the 40-bit secret key was cracked.

If you have an FTP daemon, overall you will not be compromising data security by sharing directories between this daemon and your Web daemon. However, no remote user should be able to upload files that can later be read or executed by your Web daemon. Otherwise, a hacker could, for example, upload a CGI script to your ftp site and then use his browser to request the newly uploaded file from your Web server, which could execute the script, totally by-passing security! Therefore, limit ftp uploads to a directory that cannot be read by any user.

## Understanding and Estimating the Threat

That there is real danger to information assets and systems is beyond question. No one who reads the newspapers or pays attention to the other journalistic media can have missed such stories as the denial of service attacks against the Internet (November 2, 1988). Or noted the net attack on Citibank that allegedly resulted in US\$ 2.8 million in illicit funds transfers, although Citibank claims that only about US\$ 400,000 was not recovered (Reuters, August 18, 1995). Or read Cliff Stoll's fascinating description in his book *The Cuckoo's Egg* (1989) of the tracking and capture of German hackers funded by the KGB to break into United States Government computers.

### *(d ) The Virus Threat*

Consider the problem of computer viruses. It is estimated (by the National Computer Security Association in 1996) that there are some 8,000 or more viruses in circulation and that 71 percent of all corporate networks have been or are infected (Communications Week, September 18, 1995). Viruses are so pervasive that they have been detected in shrink-wrapped software shipped directly from the manufacturer. New ones crop up at a rate that exceeds twenty per week. The question is not so much "Will you get a virus?" as "When will you get a virus?" But so what? Why not just get an anti-virus software package?

Viruses are just programs, of course, and so can be detected by looking for characteristic sequences of instructions that comprise either the part of the program that makes copies and sends them along to spread the infection or the part that does the dirty work - the payload - that displays an annoying message or destroys your data. And therein lies a problem. The anti-viral software has to have been taught to

recognize the instruction string, or "signature," of the virus in order to be able to detect it. A new virus will not be detected at all unless, as occasionally happens, the programmer just used an old virus and changed the payload. That's what happened when the "Stoned Virus" that displayed a message recommending legalization of marijuana was mutated into "Michaelangelo" that destroys data on March 6th, the painter's birthday. Of course, anti-viral software can be updated as new viruses or new versions of old viruses are discovered, but it's always a game of catch-up ball, and even those who take care to upgrade often will not be completely safe.

Moreover, the programmers who create viruses, keep up with the state of the art in anti-viral software, and constantly improve their malicious technology. We are now seeing viruses that are encrypted to escape detection. Other viruses use compression technology to make transmission easier and recognition more difficult.

Since the order in which instructions are executed can sometimes be changed without changing the ultimate result, as when two processes are independent and either can run first, the order of the instructions in a virus may be changed and thereby invalidate the anti-viral software. Or NULOPS, instructions to the computer to do nothing for a clock cycle, might be inserted at random points, mutating the sequence of instructions for which the anti-viral software seeks. Such changes result in viruses that are called "polymorphic" because they constantly change the structural characteristics that would have facilitated their detection. And lately, we have begun to see foxy viruses that recognize that anti-viral software is a work, watch as sectors of the storage device are cleared, and copy themselves over to previously cleared sectors, in effect leaping over the anti-viral bloodhound. Thus, while anti virus packages are a valuable, even essential, part of a sound information security program, they are not in and of themselves sufficient. Good backup procedures and sound policies designed to reduce the likelihood of a virus attack are also necessary.

Sound security policies, practices and procedures like those discussed in this chapter can reduce the risk they represent to a manageable level. Much more dangerous are the risks posed by directed threats, capable and willing adversaries who target the confidentiality, integrity and availability of our information assets and systems.

## ***Outside Threats***

Hackers have received a great deal of attention in the press and in the entertainment media. They comprise an interesting subculture, technically astute and talented even if socially and morally deprived. They have been pictured as nerd teenagers who stay up all night eating pizza, drinking sodas as they crouch over their computers, monitors reflecting off of their bottle-thick eyeglasses, and try command after command until they get through their school's computer security so they can improve their grade point average.

If this representation was ever accurate, it certainly is not so today. Today's cyberpunks may mostly be yesterday's juvenile cyberdelinquents grown older, but they tend to be in their twenties and even thirties, although the occasional teenager is still arrested for hacking. To the extent that hackers have a coherent philosophy, it centers around the quaint notion that, "Information wants to be free." The hacker philosophy is libertarian, and technocentric. Access to computers and information, they believe, should be unlimited and hackers should be judged solely by their computer and network skills, not by archaic

laws and ethics the evolution of which have not kept pace with the revolution in technology.

When outside hackers have the resources of a large company or a government behind them, they become even more dangerous. Large companies and governments can afford to apply resources to cracking our systems and networks that individuals would have trouble marshaling, including off-the-shelf equipment like supercomputers or arrays of general purpose computers or such special purpose devices as Field Programmable Gate Arrays. Boards are readily available with FPGA chips that can test 30 million DES keys per second at a cost about ten percent of the cost of a PC. For companies and governments, investments in custom-made special purpose chips are feasible that accelerate calculations and make the cost per solution much lower. For an investment easily within reach of a large company or a small government, 200 million DES keys could be tested per second using Application-Specific Integrated Circuits.

Such resources change the difficulty of brute-force attacks on passwords and other access controls from practically impossible to merely time consuming, and with enough resources to trivial. Using an FPGA chip at an investment of a few hundred dollars, a 40-bit key (the maximum size for which export approval can easily be obtained) could be recovered in an average time of about five hours. An investment of a few tens of thousands of dollars could reduce the time to break a 40-bit key to a few minutes. A few hundred thousand dollars would buy the capability to break a 40-bit key in a few seconds, and a few million dollars would reduce the time to less than one second. Custom chips could easily be designed for a few million dollars that would permit 40-bit key recovery in a few thousandths of a second.

DES keys of 56-bits are more secure, of course, than 40-bit keys, but an investment of a few hundred thousand dollars could yield DES keys in a few hours and an investment of a few million dollars would reduce recovery time to minutes.

## ***Inside Threat***

Where high-quality information systems security mediates information transactions across the boundary that separates an organization's systems and networks from the lawless Cyberspace outside and protects the confidentiality, integrity and availability of the organization's information assets and systems, it may be easier and cheaper to subvert or suborn an employee than to mount a direct attack. Or the attacker may seek employment and the authorized access that follows in order to better position himself to mount a wider attack that exceeds the access that has been granted as a condition of employment.

Our defenses are mostly directed outward. Few systems have a plethora of internal firewalls mediating information transactions within the organization. Many systems provide the capability to monitor and audit information transactions, even those totally within the system.

But looking for an insider abusing privilege among the vast number of transactions taking place routinely on the system is a daunting task, and impossible without computer-based audit reduction and analysis techniques. So most of our problem lies inside. Techniques are described in later chapters for abating the resulting risks include good use of computer science and cryptography to protect information assets and systems, monitoring and auditing to detect intrusions from without or abuses by insiders, and an effective capability to react to security relevant incidents, correct problems and resume safe operations. But effective and efficient security begins with and depends upon having the proper security policies in place.

# A Word About Security Holes

Security holes can be one of the major threats your security policy should cover, especially because many of them are not stopped by the use of a firewall. The following is a list of the four types of security holes threatening the security of your network and company's assets:

- **Physical Security Holes** - Caused by giving unauthorized persons physical access to the machine, where this might allow them to perform things that they shouldn't be able to do.
- **Software Security Holes** - Caused by a bug in the application's code, or "privileged" software, which can be compromised into doing things which they shouldn't. The most famous example of this is the "sendmail debug" hole which would enable a cracker to bootstrap a "root" shell. This could be used to delete your filestore, create a new account, copy your password file, anything.

Security holes are hard to predict, spot and eliminate. The following is a small list of suggestions on how to avoid or be prepared for them:

- If you're running a UNIX server, try to structure your system so that as little software as possible runs with root/daemon/bin privileges, and that which does is known to be robust.
- Subscribe to a mailing list which can get details of problems and/or fixes out to you as quickly as possible, and then make sure to run all the patches that becomes available, as soon as they become available.
- Don't install or upgrade any system or service, unless you're sure you need it. Otherwise, you may be loading something for a hacker to use. Many packages include daemons or utilities which can reveal information to outsiders. For instance, AT&T System V Unix' accounting package includes acctcom, which will, by default, allow any user to review the daily accounting data for any other user. Also, several TCP/IP packages automatically install/run programs such as rwhod, fingerd, and tftpd, all of which can present security problems.
- Don't trust installation scripts. Many of them tend to install/run everything in the package without asking you. Thus, check the list of programs included in the package before beginning to install it.
- Watch for security holes generated by incompatible usage of hardware and software. Many times, due to lack of experience, an administrator can install a software in a hardware where compatibility issues exists, which is capable of generating serious security flaws. It is the incompatibility of trying to do two unconnected but useful things which creates the security hole. Problems like this are very difficult to detect once a system is set up and running, so it is better to build your system with them in mind.
- Choosing a suitable security philosophy and maintaining it.

As Gene Spafford (spaf@cs.purdue.edu) commented on the USENET once, there is a "fourth kind of security problem [which] is one of perception and understanding. Perfect software, protected hardware, and compatible components don't work unless you have selected an appropriate security policy and turned on the parts of your system that enforce it." And he continues... "having the best password mechanism in the world is worthless if your users think that their login name backwards is a good password! Security is relative to a policy (or set of policies) and the operation of a system in conformance with that policy."

To find security holes, and identifying design weaknesses it is necessary to understand the system control structure, and layers. In order to do that, you should always try to:

- Determine the items to be protected, or security objects, such as users file.
- Identify your control objects, or the items that will protect security objects.
- Detect potential holes in a system. These holes can often be found in code that:
- Is ported to a new environment.
- Receives unexpected input.
- Interacts with other local software.
- Accesses system files like passwd, L.sys, etc.
- Reads input from a publicly writable file/directory.
- Diagnostic programs which are typically not user-proofed.
- Test code for unexpected input. Coverage, data flow, and mutation.

I hope all the above gave you an idea of what a security policy should contain. Vulnerabilities are many, Internet attacks as well. There are several countermeasure strategies you can use, but without a guideline, a map, you might find yourself shooting in the dark. That's when a security policy is necessary. It will become your map, your guideline, your contract (with users and upper management), your "power of attorney" to take the decisions you must take in order to preserve the security of your site.

## Setting up a Security Policy

As discussed earlier, an Internet firewall does not stand alone--it is part of the organization's overall security policy, which defines all aspects of its perimeter defense. To be successful, organizations must know what they are protecting. The security policy must be based on a carefully conducted security analysis, risk assessment, and business needs analysis. If an organization does not have a detailed security policy, the most carefully crafted firewall can be circumvented to expose the entire private network to attack.

The following is a template of a typical security policy. Use it as a foundation for your own security policy, add and remove whatever doesn't apply to you and make sure to have as many inputs as possible from upper management, which should totally support it.

## *A Security Policy Template*

<Your Company> INTERNET SECURITY POLICY

### **1 PURPOSE**

This regulation establishes minimum security requirements for the use of the Internet network by <Your Company>. This regulation is not written to restrict the use of Internet, but to ensure that adequate protection is in place to protect <Your Company> data from intruders, file tampering, break in, and service disruption.

### **2 BACKGROUND**

In the late 1960s the Department of Defense (DoD) designed and implemented the ARPAnet network for the exchange of defense industry research information world-wide. TCP/IP was the protocol developed and UNIX was the platform.

The National Science Foundation (NSF) needed a network also to interconnect their supercomputers and exchange academic research information so they built their own, but followed the DoD standards. They called their network NSFNET.

The Internet consists of many, worldwide, independent networks that allow interconnection and transmission of data across the networks because they follow the same basic standards and protocols and agreed upon Internet etiquette, " No central authority." Each user organization pays for its own piece of the network.

Motivated by developments in high-speed networking technology and the National Research and Education Network (NREN) Program, many organizations and individuals are looking at the Internet as a means for expanding their research interests and communications. Consequently, the Internet is now growing faster than any telecommunications system thus far, including the telephone system.

New users of the Internet may fail to realize, however, that their sites could be at risk to intruders who use the Internet as a means for attacking systems and causing various forms of threat. Consequently, new Internet sites are often prime targets for malicious activity including break in, file tampering, and service disruptions. Such activity may be difficult to discover and correct, may be highly embarrassing to the organization, and can be very costly in terms of lost productivity and compromised data integrity.

All Internet users need to be aware of the high potential for threat from the Internet and the steps they should take to secure their sites. Many tools and techniques now exist to provide sites with a higher level of assurance and protection.

<Your Company> branches should acquire a copy of the "Guide to the <Your Company> Internet." This document is published by the MIS department. This guide defines the <Your Company> Internet Access Network. You may acquire this guide by contacting the Director of MIS, at extension XXX.

### **3 DEFINITIONS**

Definitions relating to this policy may be found in appendix "A".

### **4 REFERENCES**

NIST CSL Bulletin, July 1993, NIST Connecting to the Internet: Security Considerations

<List here any other documents users can refer to in order to better understand this policy>

### **5 ABBREVIATIONS**

**ARPAnet** Advanced Research Projects Agency Network

**DMZ** Demilitarized Zone

**DoD** Department of Defense

**FTP** File Transfer Protocol

**LAN** Local Area Network

**NIST** National Institute of Standards and Technology

**NSF** National Science Foundation

**NFS** Network File System

**NREN** National Research and Education Network

**OSI** Open System Interconnect

**TCP/IP** Transmission Control Protocol/Internet Protocol

**TCP** Transmission Control Protocol

## **6 POLICY**

The responsibility for protecting <Your Company> resources on the Internet is the responsibility of the <Your Company> or Staff Offices. This policy apply to contractors and universities that connect to <Your Company> computer. <Your Company> branches which access the Internet must develop and implement an Internet security policy which meets the minimum requirements of this regulation as following:

1. Data which is exempted from disclosure under the Freedom of Information Act (Public Law 93-502) or whose disclosure is forbidden by the Privacy Act (Public Law 93-579) will not be transmitted over the Internet network unless encrypted. "Note: Logon IDs and passwords are frequently classified as sensitive information."
2. All <Your Company> branches and staff offices using the Internet must follow the guidance in <any additional documentation>.
3. <Your Company> branches and staff offices that plan a gateway to the Internet are responsible for funding, implementing and maintaining the prescribed protection, including devising, and implementing a comprehensive risk management program.
4. Agencies and staff offices will access the Internet only through the <Your Company> Internet Access Network.
5. Host-based security will be the primary method of protecting <Your Company> systems. However, many host-based security software packages cannot be trusted to protect us from the Internet, because of their vulnerability to denial-of-service attacks.
6. Due to inherent weaknesses in certain Internet telecommunication services, and cumbersome aspects of some security packages, many sites will find that the most practical method of securing access to systems from the Internet is to use a secure gateway or a firewall system. <Your company> branches and departments will perform risk assessments to determine where secure gateways, firewalls, smart cards, or authentication tokens will be most suitable. <Your Company> branches will:
  - Use firewalls and/or packet filters on the local routers, when the system uses TCP/IP.
  - Configure firewalls on with outgoing access to the Internet, but strictly limit incoming access to <Your Company> data and systems by Internet users.
  - Apply the DMZ concept as part of the firewall design.



- Firewall compromise would be potentially disastrous to subnet security. For this reason, branches will, as far as is practical, adhere to the below listed stipulations when configuring and using firewalls.
- Limit firewall accounts to only those absolutely necessary, such as the administrator. If practical, disable network logins.
- Use smartcard or authentication tokens to provide a much higher degree of security than that provided by simple passwords. Challenge-response and one-time password cards are easily integrated with most popular systems.
- Remove compilers, editors, and other program development tools from the firewall system(s) that could enable a cracker to install Trojan horse software or backdoors.
- Do not run any vulnerable protocols on the firewall such as TFTP, NIS, NFS, UUCP.
- Consider disabling finger command. The finger command can be used to leak valuable user information.
- Consider not using the e-mail gateway commands (EXPN and VFRY) which can be used by crackers to probe for user addresses.
- Do not permit loopholes in firewall systems to allow friendly systems or users special entrance access. The firewall should not view any attempt to gain access to the computers behind the firewall as friendly.
- Disable any feature of the firewall that is not needed, including other network access, user shells, applications, and so forth.
- Turn on full-logging at the firewall and read the logs weekly at a minimum.
- No <Your Company> computer or subnet that has connections to the Internet can house privacy or sensitive information without the use of firewalls or some other means to protect the information.
- <Your Company> branches and staff offices must develop and document an Internet security strategy based on the type of Internet service selected for use. This strategy must be included in the Internet Security Plan.
- <Your Company> branches and staff offices that use the Internet must adhere to guidance stated in XXXXX" <Your Company> Internet Security Policy."
- All software available on the Internet must be scanned for Trojan horses or computer viruses once it has been downloaded to a <Your Company> computer.
- All downloaded software should be loaded preferably onto a floppy disk and not to the system hard disk. Once you are reasonably assured that the downloaded software does not contain Trojan horses or computer viruses it can be placed on the hard drive. If the software will not fit on a floppy disk then the only option is the hard disk. The software must be scanned before use (executed).
- Mandatory vulnerability and risk assessment of existing gateways is required at annual intervals. Initial assessment should be completed within nine (9) months of the issuance of this policy. And all branches should also conduct weekly or monthly reviews of audit trails of gateway software and firewalls for breaches of security.
- <Your Company> personnel, and contractor personnel working for <Your Company> while using the Internet:

- Must not be harassing, libelous, or disruptive to others while connected to the Internet.
- Must not transmit personal data or unauthorized company-owned data across the Internet.
- Must obey all copyright laws.
- Must not download to company's computers from the Internet any obscene written material or pornography.
- Must not send threatening, racially harassing, or sexually harassing messages.
- Must not attempt to break into any computer whether <Your Company>, its clients or private.
- Must not be used for private or personal business, except when authorized.
- Must not introduce computer viruses, worms, or Trojan horses.
- <Your Company> sponsored Internet connections are to be used for official <Your Company> business.
- Host computers should be regularly scanned to ensure compliance with <Your Company> security guidelines.

## 7 RESPONSIBILITIES

The Director or MIS :

1. Develop, coordinate, implement, interpret, and maintain Internet Security policies, procedures, and guidelines for the protection of <Your Company> information system resources.
2. Review <Your Company> Internet security policy.
3. Assist in <Your Company's> branch Internet security policy development and implementation.
4. Determine adequacy of security measures for systems used as gateways to the Internet.
5. Ensure that all <Your Company> branches conduct periodic information systems security risk assessments, security evaluations, and internal control reviews of operational <Your Company> Internet gateways and facilities.

All branches and <Your Company> departments that have or are planning to install a firewall or any sort of gateway to the Internet will:

- Devise and implement a comprehensive risk management program which assures that security risks are identified, considered, and mitigated through the development of cost effective security controls. The risk management system will include a service access policy that will define those services that will be allowed or explicitly denied from the restricted network, how these services will be used, and the conditions for exception to this policy.
- Another part of this risk management system will be a firewall design policy. This policy relates precisely to firewalls and defines the rules used to implement the service access policy.
- Each branch and staff office must develop an Internet Security Plan which address all security controls in place or planned.
- These controls shall be commensurate with the risks identified in the risk analysis. Internet Security plans shall be submitted annually with the <Your Company's> security plans for review and approval. The guidelines governing the submission of these security plans should comply to the Internet Security Plan.

- Perform risk analysis to identify the risks associated with using Internet both for individual users and branches or departmental offices. Cost effective safeguards, identified in the risk analysis process, will be implemented and continually monitored to ensure continued effectiveness.

<Your Company> MIS department should be responsible for developing, testing, and maintaining Internet contingency plans. The risk involved with using the Internet makes it essential that plans and procedures be prepared and maintained to:

- Minimize the damage and disruption caused by undesirable events; and
- Provide for the continued performance of essential systems functions and services.
- Develop, install, maintain, and regularly review audit trails for unusual system activity.
- Fund, implement, and maintain the prescribed protective features identified as a solution by a risk assessment.
- Risk assessment developed by branches and staff offices are to be made available to MIS upon request.
- Ensure that the branch information security manager is a vital part of any security activity on the Internet.

The information security manager is responsible for:

1. Implementing the policy stated in this directive.
2. Developing audit trails for any <Your Company> network connected to the Internet.
3. Reviewing and monitoring activity audit trails on the Internet connections.
4. Working closely with the branch network administrator in monitoring activity on the use of their host and subnets.

## 8 NON-COMPLIANCE

All users of data and systems are responsible for complying with this Internet systems security policy, as well as procedures and practices developed in support of this policy.

Anyone suspecting misuse or attempted misuse of departmental information systems resources is responsible for reporting such activity to their branch or staff Office management, or to the information system security manager or the MIS manager.

Violations of standards, procedures, or practices in support of this policy will be brought to the attention of management for action, which will result in disciplinary action up to and including termination of employment.

## 9 SOURCE OF INFORMATION

1. MIS Guide To The <Your Company> Internet
2. <Whatever documents you want to make available to users>

Orders

Orders

Back

Forward

|       |          |
|-------|----------|
| Comme | Comments |
|-------|----------|

---

[COMPUTING MCGRAW-HILL](#) | [Beta Books](#) | [Contact Us](#) | [Order Information](#) | [Online Catalog](#)

---

HTML conversions by [Mega Space](#).

This page updated on December 05, 1997 by [Webmaster](#).

[Computing McGraw-Hill](#) is an imprint of the [McGraw-Hill Professional Book Group](#).

Copyright ©1997 [The McGraw-Hill Companies, Inc.](#) All Rights Reserved.

Any use is subject to the rules stated in the [Terms of Use](#).

|        |        |
|--------|--------|
| Orders | Orders |
|--------|--------|

|      |         |
|------|---------|
| Back | Forward |
|------|---------|

|          |          |
|----------|----------|
| Comments | Comments |
|----------|----------|

© 1997 The McGraw-Hill Companies, Inc. All rights reserved.  
Any use of this Beta Book is subject to the rules stated in the [Terms of Use](#).

---

## Chapter 10

# *Putting It Together: Firewall design and Implementation*

This chapter discusses what you need to know about firewalls and its implementations. In some ways it complements chapter 7, "What is an Internet/Intranet Firewall After All?," as it goes beyond the basic concepts discussed on that chapter. This chapter reviews the different firewall technologies used today, their strengths and their weaknesses, and the tradeoffs involved when designing a firewall system and implementing it to your specific application and corporate needs.

## Reviewing the Basics

We discussed on chapter 7 that firewalls helps to protect private networks from unauthorized intruders. But there are many firewalls available on the market, as someone said once, "from the basement-brewed firewalls to the me-too firewalls from the larger manufacturers." Chapter 14, "Types of Firewalls and Products on the Market" gives you an extensive list of all the major firewall vendors, their products and in-depth details and discussion about the product strength, pros and cons. When reviewing those products, keep in mind that the underlying technology used in the firewalls is very important to the security and integrity of the firewalls.

As you already know, unless you skipped chapter 7, which I recommend to get back there, there are currently two main firewall technologies: packet filtering and application level. But we also discussed that depending on the technology employed, firewalls can be classified in four categories:

- **Packet filters** - This type of firewall provide access control at the IP layer and either accept, reject or drop packets based mainly on source, destination network addresses and the type of applications. Packet filtering firewalls provide a simple level of security at a relatively inexpensive price. These type of firewalls also provide a high level of performance and are normally transparent to the users.
- **Weaknesses of packet filtering firewalls:**
  1. They are vulnerable to attacks aimed at protocols higher than the network level protocol, which is the only level they understand.

2. Since the network level protocol requires certain knowledge of its technical details, and not every administrator have them, packet filtering firewalls are usually more difficult to configure and verify, which increases the risks for systems misconfigurations, security holes and failures.
  3. They cannot hide the private network topology and therefore expose the private network to the outside world.
  4. These firewalls have very limited auditing capabilities, and as you know, auditing should play a major role in the security policy of your company.
  5. Not all Internet applications are supported by packet filtering firewalls.
  6. These firewalls not always support some of the security policies clauses such as user-level authentication and time-of-day access control.
- **Application-level firewalls** - Application-level firewalls provide access control at the application-level layer. Thus, it act as application-level gateways between two networks. Since application level firewalls function at the application layer, they have the ability to examine the traffic in detail, making them more secure than packet filtering firewalls. Also, this type of firewall are usually slower than packet filtering due to their scrutiny of the traffic. Thus, to some degree they are intrusive, restrictive and normally require users to either change their behavior or use specialized software in order to achieve policy objectives. Application-level firewalls are thus not transparent to the users.
  - **Advantages of application-level firewalls:**
    1. Since they understand application-level protocol, they can defend against all attacks.
    2. They are usually much easier to configure then packet filtering ones, as they don't require you to know all the details about the lower level protocols.
    3. They can hide the private network topology.
    4. They have full auditing facilities with tools to monitor the traffic and manipulate the logs files which contain information such as source, destination network addresses, application type, user identification and password, start and end time of access, and the number of bytes of information transferred in all directions.
    5. They can support more security policies including user-level authentication and time-of-day access control.
  - **Hybrid firewalls** - Realizing some of these weaknesses with packet filtering and application-level firewalls, some vendors have introduced hybrid firewalls which combine both packet filtering with application-level firewall techniques. While these hybrid products attempt to solve some of the weaknesses mentioned above, they introduce some of the weaknesses inherent in application-level firewalls as outlined above.
  - **Weakness of hybrid firewalls:**
    1. Since hybrid firewalls still rely on packet filtering mechanisms to support certain applications, they still have the same security weaknesses
  - **Second-generation application-level firewalls** - This type of firewall is still being an application-level firewall, only that in its so called second generation, which solves the transparency problem of its earlier version without compromising performance.

- **Advantages of second generation application-level firewalls:**

1. They can be used as an Intranet firewall due to their transparency and generally higher performance.
2. They can provide full network address translation in addition to network topology hiding.
3. They can support more advanced user-level authentication mechanism.

## Selecting a Firewall

Before you start selecting a firewall from chapter 14, you should develop a corporate security policy, as discussed on chapter 7, and then select the firewall that can be used to implement the chosen policy. When evaluating firewalls, care must be taken to understand the underlying technology used in the firewall as some firewall technologies are inferior in security to others.

The basic concept of a firewall will always be the same, so you should evaluate a firewall based on the level of security and implementation features it offers. When I say security features I mean the ability a firewall product has to deliver security based and consistent to your corporate security objectives and policy. The following are some of the characteristics you should be looking for in a firewall:

- **Security Assurance** - Independent assurance that the relevant firewall technology fulfills its specifications and assurance that it is properly installed. Is the firewall product certified by the National Computer Security Association (NCSA - <http://www.ncsa.com/>) What about the Communications Security Establishment (CSE) evaluation, does it have one?
- **Privilege Control** - The degree to which the product can impose user access restrictions.
- **Authentication** - What kind of access control the product provides? Does it support authorizations? What about authentication techniques. These techniques include security features such as source/destination computer network address authentication, password authentication, access control cards, and fingerprint verification devices.
- **Audit Capabilities** - The ability of the product to monitor network traffic, including unauthorized access attempts, generate logs, and provide statistical reports and alarms.

As for implementation features, you should be looking for the ability a product has to satisfy your network management requirements and concerns. A good firewall product should be:

- **Flexibility** - The firewall should be open enough to accommodate the security policy of your company, as well as allow for changes in the future. Remember, a security policy should very seldom change, but security procedures should always be reviewed, especially in light of Internet and Web-centric new applications!
- **Performance** - A firewall should be fast enough so that users wouldn't feel the screening of packets. The volume of data throughput and transmission speed associated with the product should be reasonable enough, consistent to your bandwidth to the Internet.
- **Scalability** - Is the firewall scalable? The product should be able to adapt to multi-platforms and instances within your protected network. This includes OSs, machines and security configurations.

As far as integrated features, look for the ability of an firewall to meet your and users needs, such as:

- **Ease of Use** - The firewall product should ideally have a Graphical User Interfaces (GUI), which

simplifies your job when installing, configuring and managing it.

- **Transparency** - How transparent is the firewall product to your user? If you adopt a confuse system to use, then the users will develop resistance against it and will end up not using it. Conversely, the more transparent the firewall is to your users, the more likely it will be for them to support you and use it appropriately.
- **Customer Support** - The extent to which a vendor supports customer needs, such as providing prompt access to technical expertise for installation, use and maintenance, and comprehensive training courses.

## Considerations About the Security Policy

A security policy is very important when setting up a firewall at your company, as it outlines what assets you consider worth protecting and what actions or risk management procedures you must cover in order to protect your corporate assets.

Network security policies often must integrate security issues from all previous policies. Usually companies seek outside assistance when first creating their network security policy.

The following is a boiler plate for you to use when creating a security policy. Make sure to add or remove any item that don't apply to your environment:

### "Your Company Name" Security Policy

#### I. Security Policy

Definition

Reasons for adopting a security policy

Mission statement

#### II. Security Policy and Procedures

How should it be reenforced

Support from upper management

Special circumstances and exceptions to the rule

Need for upper management approval

#### III. Development of General Security Policy

Objectives and security goals

Definitions

General security policy and procedures:

About the networks

About the Intranet



About the Internet

About the Extranet

About telecommuters

About remote users

About Application use

About hardware use

#### IV. Security Profile

Of desktops and workstations

Of networks

Of Intranet

Of Internet

Of Extranet

Of applications

Of telecommuters

Of remote users

#### V. Profile of Threats and Countermeasure

Viruses

Worms

Applets

Trojan Horses

Security holes

Espionage

#### VI. Developing Specific Rules and Procedures

For the company

For personnel

For wiring

For networks

For logistics

For Operations

For workstations

For servers

For remote access services

## VII. Technical Support

Common goals and mission statement

Specific goals and procedures

Procedures for auditing corporate security

## VIII. Auditing Policy

Automatic generation of login reports

Security checklist

## IX. Technology Policy and Procedures

Adopted access control mechanisms

Firewall and proxy servers

Security management

Risk management and control

# ***Issues to Consider About Physical Security***

Network security interacts with physical security because the size or shape of the network "machine" or entity can span a building, campus, country or the world due to interconnections and trust relationships. The weakest link in an international network, for example, may be the fact that a serial-line maintenance cable passes over a public restroom at corporate headquarters! Physical security policy may have to be updated, and the physical policy must be taken into account when creating the network policy.

# ***Issues to Consider About Access Control***

Access control explicitly decides for each packet of network traffic whether or not it is allowed and what action is appropriate. A firewall determines if the packet or session is consistent with it's copy of the security policy.

With a sufficiently powerful policy engine, the firewall can implement fine-grained (and therefore more secure) policies. Good policy engines impose the fewest restrictions on which policies they can implement. Good access control includes managing remote access and enables administrators to provide more or less access to users depending on from where they are working.

## ***Issues to Consider About Authentication***

Authentication is how users to the network infrastructure who they are. The type of authentication used varies depending on from where users are authenticating. From their desk, a simple user id and password may be sufficient because of the accompanying physical security. When connection to the firewall from the Internet, a token-based authentication may be necessary.

## ***Issues to Consider About Encryption***

Encryption can ensure data integrity or protect sensitive information sent over insecure lines. Such protection is usually essential for remote access to important company assets or as extra protection when using company Intranet.

A serious issue with encryption is how to manage the "keys." Keys are used to encrypt and decrypt the data. If you have only one or two connections that must be encrypted, then manual key distribution is fine. If you have 100's or 1,000's of keys to distribute, then only automated key management will work. Manual distribution of large numbers of key is too insecure or costly.

## ***issues to Consider About Security Auditing***

Once a security policy has been implemented, it must be periodically checked to ensure that all components and employees are in compliance. Without sufficient auditing, a company may have no legal recourse if there is a security breach. Auditing can also find problems before they turn into security breaches. Auditing and monitoring products are relatively new so many tasks must still be done manually and less frequently than desired.

## ***Issues to Consider About Training***

You must train your users about the information system in place, from the desktop, and applications to the network and access to the Internet. Otherwise they will be one of the most serious threat to your network security. If your users do not understand the power and proper use of your network, then they can unintentionally compromise security. In particular, employees must manage passwords properly and recognize when someone asks them for inappropriate information about the network.

## **Responding to an Incident: Your Network Under Attack**

It is very hard for you to detect if your site has been broken-in to. If your site was been broken by a hacker, chances are you would never know! They are very difficult to be detected! If it was broken by a cracker, you may be able to trace it more easily.

Fortunately, for those of you using an UNIX systems, there is a program called "tripwire," that can

perform periodical scans in your system to detect if any system files or programs have been modified. But this is not enough to prevent a hacker to invade your system and not every operating system platforms have tools like tripwire.

**Tip:**

Tripwire is distributed free of charge. If interested in downloading it, try URL:  
<ftp://coast.cs.purdue.edu/pub/COAST/Tripwire/>

Another quick check you can do is to check your access and error log files for suspicious activity. Look for traces of system commands such as "rm", "login", "/bin/sh" and "Perl."

For those on Windows NT platform, check the Security log in the Event Log periodically, looking for suspicious activities.

Also, hackers usually try to trick a CGI script into invoking a command by entering long, very long lines in URL requests with the purpose to overrun a program's input buffer. Lastly, look for repeated failed attempts to access a password or section protected by passwords. Overall, these could be an indication that someone is trying to break-in to your site.

Sites are been broken-in to more and more every day. As the technology, specially the Web technology, changes so rapidly, systems become obsolete or vulnerable to new threads very quickly. Even the most protected system can become vulnerable by a creation of a new Java applet not predicted into its present security system.

Web server running operating systems (OS) such as SunOS and UNIX, which are based on the client/server abstraction are particularly sensitive to these moving Internet technology trends. As they usually are developed to model the network as an extension of its internal data bus, which also extends a series of features hardly found in other OS platforms, these same extensions opens a door (if not many!) to hacker and intruders.

But opening a door for potential hackers is only the index of the whole Bible! There is much more to it when come to intrusion detection.

Nowadays, hackers are very aware of the typical security models utilized by MIS and deployed all over the Internet. As a matter of fact, they use it for they own interest. Password systems, access and authentication systems are not sufficient to guarantee the security of a protected network.

Hackers can write simple applets to act as Network File System (NFS) clients, for instance, and bypass all the access control system normally used, gaining total access to internal networks or users files. But this is not merely a security hole of NFS, it extends to almost every network service available.

When comes to secure your site, you must rely and apply every resource you can to guarantee the safety of your site and users. Firewalls and proxy server, as you saw on chapter 10, will not totally resolve the problem, but they will greatly enhance your chances of survival.

Once you have done everything you can to protect your site, from hardware to software, from security policy to its implementation, the only thing you can do is to accept the odds and wait for the day that you

may need to face an incident. I hope you never will, but if you do, you must be prepared to deal with the incident, from the systems perspective, to the legal one.

## ***Dealing With an Incident***

Our battle in protecting our site and protected network system can be compared with the same battle we go through in protecting our own bodies. In order to prevent from a virus disease we must isolate it, analyze it, observe it, learn from it, so that we can reverse its vital conditions and hopefully exterminate it.

The same is true with computer security. We must be able to isolate our attackers, analyze it, observe it and learn from it as well. Unfortunately, not much is discussed about the hacker, the intruder, the attacker. Much description goes for the symptoms that evidences its presence and the devastating consequences of its attacks.

Just like with viruses, no one really knows much about the hackers, only about the signs of their presence. The good news is that this hacker do have forms. Most of them are usually male, computer science students. Of course, all of them have access to the Internet and know very well the UNIX environment.

We could move on and ask ourselves why the hacker do what they do. But it would be out of the scope of this book, so lets just say, I hacker likes to challenge himself and the majority of the times he will break into a system, just for the challenge of been able to do so.

A hacker, like a bug (or virus), usually follows a standard pattern to break into a site. Usually he tries to:

1. **Determine his next target**, which is the systems he will be working on. This is usually accomplished by checking machines listed in .rhosts and .netrc files, found on systems that already were broken in to. Also, a hacker can try to gather such a list from the domain name system, or DNS. Through the DNS a hacker is capable to know the machine name, its Internet Address, the type o machines, and even the owner and department that machine belongs to.
2. **Access the target system**, which will require from a hacker to forge his i.d. as of a regular user within the company. In systems where the authentication of an user relies on his username and password, a hacker has some advantage as usernames are usually known by everyone.

Many times these usernames are composed of the last name and first initial or a combination of both. Even if a username is not so obvious, it is easy to obtain through finger and ruser.

However, the password is not something so easy to break if users choose them with at least 12 to 15 characters, and they are not found on any dictionary. There are too many combinations to be tried, and even if a hacker is to use password cracking tools (see Appendix H), it is too much time consuming and not guaranteed to work in a timely fashion. At least, it will demand patience and a lot of time. Further, with Windows NT and the majority of the operating systems, the machine will disconnect after the third or fifth attempt to enter a correct password anyway. That is why a hacker will usually rely on network services such as NIS, RLOGIN/RSH, NFS. More will be discussed about it ahead.

3. **Consolidate his position**, by using regular services as cracking tools.

There are several services that can be used as cracking tools as well. The following is a selection of some of them, which you should be careful when making them available and using them.

## ***Network Information Service as Cracking Tool***

The network information service (NIS) can be very useful for a hacker as it provides a database service of multiple clients and replicated servers. This service stores various information such as password files, group files, and the like. Only clients with the NIS domain name have access to it, as a way to protect it, since it holds sensitive information.

Unfortunately, by default installation, usually the NIS domain is also the DNS domain name for the site, or something very similar. Once a hacker gets access to it a get a copy of the password file, he only needs to run one of the many password cracker applications (some listed on Appendix H) to gain access to the system.

This is very practical with UNIX-based systems, but not so with Windows NT, Macintosh and other platforms. However, as I write this book couple password cracking systems were released for Windows NT.

In trying to resolve this problem, shadow passwords can be used. This consists of two databases, the actual password file, available for privileged users and another one available to everyone. It makes it harder to break in but not impossible as a shadow file can still be read by anyone as long as they make the request from a privileged port.

## ***Remote Login/Shell Service as Cracking Tool***

The remote login service provides a remote terminal service so that users can access the network remotely as if they were directly attached to it. It is much the same of the Remote Access Service (RAS) for Windows NT and Windows 95. The authentication in these services is done by entering the username and the password, as well as the domain name.

The problem is that many times, in the attempt to ease the operation for the users, the domain name and password is giving automatically. The user only needs to enter his/her password.

Also, the user usually has an option to have the password remembered by the system next time he/she logs in so it doesn't have to be entered! The remote shell service, a related service, allows anyone already authenticated into the system to log in onto trusted domains and execute commands on those domains. This service uses the same authentication mechanism as the login service; for this reason the two services are discussed here as one. If not monitored very closely, remote access sessions can also be a major threat to the system, as very often it even by-passes any installed firewall.

When used wisely these trust mechanisms are very valuable. Users don't have to type their password every time they log into a trusted machine and remote commands can be executed without logging in first. The bad decision here is to let the user decide who to trust and who not to trust! For example, if in a .rhosts file the user Mario trusts the user Lourdes and in turn Lourdes trust the user Marcio and Marcio trusts the user Celia, Celia can become Lourdes through the trusting relationship. If any oth this accounts are broken in to, all the other ones become exposed as well.

# ***Network File System as Cracking Tool***

Sun's Network File System (NFS) is one of the most important and vulnerable network service in the system, as it provides full access to files and directories. The major security hole is that NFS's access control mechanisms are very hard to maintain, and are hardly adequate. Another hole is that it doesn't have user authentication, even when using the so-called secure NFS implementation.

Every user can write his own NFS client, specify any identity and read or write files. An NFS client that provides this basic functionality can easily be written in about 300 lines of C code. The secure NFS tries to fix this security hole but it doesn't totally succeeds. The problem is that the underlying cryptosystem doesn't work, and can be broken very easily.

File handles also used to (it has been fixed!) represents a major vulnerability. They can be constructed without the help of the mount daemon, which allows a client to directly go to the NFS daemon, and bypass the access control mechanisms which are enforced by the mount daemon.

# ***File Transfer Protocol Service as Cracking Tool***

The File Transfer Protocol (FTP) service allows clients to copy files from one machine to another. It resembles NFS in a way, but is intended for long haul networks. Clients normally need to be authenticated.

Nevertheless, FTP implementation has been well known for its security holes. Over time, FTP has become a very complex and difficult system to understand as features were added. For instance, a major security hole of this system is that it can be tricked to give a hacker the permissions of a determined user, while the hacker actually logs in using a public account. These bugs have all been fixed, but FTP services became so broad that I recommend you to watch it very closely.

That is why it is so important to keep your eyes on the directory permissions of an FTP server. Once a hacker is in, the first thing he will check is if he can write do that directory. If so, he will probably put a .rhosts file into it containing his name and his current machine. Since the directory is often the home directory of the user ftp (or ftpd), a simple remote login sufficed to get into the system!

There are several other security risks with FTP, which is not the scope of this book to get into it. The purpose here is to show you the many open doors present in your system, even with a firewall in place.

Just like a virus, a hacker goes to a period of "incubation." Once he breaks into a system, he starts to consolidate his position, which I call the incubation stage. Usually it is done by simply placing a .rhosts file in the home directory of a cracked account. There are other methods, but this is usually very effective.

Besides consolidating his presence in this new "body," a hacker is also interested on what is there for him: mailboxes, user's information, etc. Once inside, a hacker will not waste time in consolidating his position there. Usually he will target applications that require passwords, such as TELNET and FTP.

Therefore, it is important that you try spot a hacker as soon as you can, possibly before he consolidates his presence within your system. The following is a shell script sample for spotting a hacker. You can

base on it to tailor your own, depending on the system you are trying to protect:

```
#!/bin/sh
```

```
LOGFILE=logfile
```

```
while true; do
```

```
    case 'date | cut -d" " -f5 | cut -d: -f1' in
```

```
        (18|19|20|21|22|23|00|01|02|03|04|05|06|07)
```

```
            (echo "====="; date) >> $LOGFILE
```

```
            (echo "who"; who) >> $LOGFILE
```

```
            (echo "ps axl"; ps axl) >> $LOGFILE
```

```
            (echo "netstat -n"; netstat -n) >> $LOGFILE
```

```
            sleep 600
```

```
            ;;
```

```
        *)
```

```
            sleep 3600
```

```
            ;;
```

```
    esac
```

```
done
```



# ***To Do List in Case of an Incident***

I hope you will never have a break in at your site. But unfortunately, chances are that you will. So if you ever need to respond to a security incident there are some steps worth following. Although you don't need to follow these steps in order, and maybe some of them not even apply to your situation, you should at least review them as it will help you to get control of the situation sooner, than later.

Borrowing Garfinkel's rules for incident response, make sure you:

1. Don't panic! And
2. Document everything!

Many times, in face of a hacker's attack, there is not much you can do other than feel sorry about what happened to your site and/or users. Other times, you might be able to even stop a hacker from going any further! So your steps, in case of an incident should be:

- Assess the situation,
- Cut off the link,
- Analyze the problem, and
- Take action.

## ***Assessing the Situation***

The first initiative you should make when a break in is confirmed should be to assess the damages, the seriousness of the break in as soon as possible. For instance, there was a time I used to be directly involved with computer security incidents. I used to have a white board at my office for situations like that. In every situation, I would start assessing the incident by writing down few questions on the board:

- Did the hacker actually succeeded in breaking into the site? If so, you will need to act very quickly, regardless if the hacker is still in or not. The main goal here isn't to catch him, but to protect your users, documents and systems resources.
- Is the hacker still acting in your system? If so, you will need to stop him! You will need to decide when and how, but it should be as soon as possible. If not, then you may have some time to work on it before he strikes again.
- What is the best way to halt the system until you can have more control over the situation? You may have to shutdown the system, or at list stopping the affected service (FTP, Gopher, TELNET, etc.). Maybe you will even need to shutdown your Internet connection.
- Is there any possibility for this attack to be an inside threat? If so, you will need to be more careful not to let the solution you will take transpire.
- Can you learn with it the attacker? If you think you are protected, that nothing is at stake, then you may want to give some line for the "fish" and learn a little more about it, see what he is up to.

# ***Cutting Off the Link***

Once you assessed the situation, you should be in the position to start making some decision, taking some actions. At least short term ones. The first one should be to cut off the link. What that means will depend on your environment. Go back to your white board, take a look at the notes already there and based on it put some more down:

- Can you shutdown your server? Do you need to? If you can, you might as well do it. It will give you some time to get the facts straight. If you can't, you may want to shutdown some services, or at least log everyone out.
- Do you care about tracing down the hacker? If you do, you may not want to shutdown the connections to the Internet, as you will lose track of him.
- Is it possible that other clients were affected? If so, you may want to shutdown the server and check every one directly connected to the server.
- By shutting down the server, can you afford to lose some useful systems information you may need.

## ***Analyze the Problem***

Now it is time to go back to your white board, add up all the information you wrote there, subtract the hyper reactions the followed the realization you have been hit and come up with the results. At this point, you must have a plan.

Take your time. The worse is over, your system is probably already down, and surely, you will learn something new today! Make sure to think thoroughly about the actions you are about to take. Evidently, at this point you already identified the security hole and will be fixing it. Make sure your fix won't create another security hole or will affect other services or processes. Will your approach resolve the problem?

Once you have the whole plan ready, bounce it off someone else you trust. Try someone out of the picture, not affected by the same bias you may are.

## ***Take Action***

It is time to implement your emergence response plan. Make sure upper management, users and service providers are aware of the incident.

You don't need to give them much information, especially the technical ones, but should give them a reasonable timeframe for the restoration of the system.

Notify CERT, exchange your information with them. Not only you will be helping them to alert others about it but they might be able to help you with their expertise.

Finally, repair the security hole and restore the system. Make sure to document the whole incident, learn from it and archive it.

# ***Catching an Intruder***

It is very difficult to catch intruders. Especially when they try to cover up their tracks. Chances are that if you are able to spot a hacker attack, it will be by accident! Very unlikely intentional.

However, even though you will need a lot of luck to spot a hacker in your system, there are some guidelines that you can follow to help you to be more lucky:

- Always keep an eye on your log files, examine them regularly, especially those generated by the system log service and the wtmp file.
- Watch for unusual hosts connections, as well as unusual times (instruct users about connection times, so it can be easier to eliminate possibilities).
- Watch for accounts that are not being used for a while and suddenly become active. You should always disable or delete unused accounts.
- Expect a hacker's visit usually between the hours of 6PM to 8AM, Saturdays, Sundays, and holidays. Yes, they can come at any time!
- Set a shell script to run every 10 minutes during these times logging all the process and network connections. For instance, I have a log file set in Performance Monitor (Windows NT) running during those hours, tracking RAS connections, processes and network connection activities).to a file. The shell script above is an example of it. But don't count on it. Hacker are not stupid and will quickly find out that they are been watched!

## ***Reviewing Security***

There is so much that should be discussed when reviewing security. Many books were written about it. Associations and task forces were created for that purpose.

The following is a summary list of security issues you should review. It is not a complete list, of course, but it does try to address some of the main issues affecting your Web environment. At the end of this book you will find some complementary bibliography references do complement this information:

- Make sure to install the NIS latest patches when working with it.
- Do not use any wildcards on trusted hosts databases (/etc/hosts.equiv), if you have any, remove them.
- Be very careful when using .rhosts and .netrc. You should consider disallowing them from foreign hosts.
- As with trusted hosts database, do not use wild-cards nor store plain-text passwords in the .netrc file.
- As with NIS, make sure to install the latest patches for NFS.
- Ensure that you specify to which hosts you export your file system. You may want to write-protect the user file system (/usr) when exporting it.
- You may want to disallow setuid and root access for any NFS file system.
- You may also want to turn off the off the "-n" option of the mount daemon (/etc/rc.local).  
Although some people believe the system will be slightly unsecured (including the mount daemon

manual), this is not true. You will have not security at all!

- When offering FTP service, make sure to write-protect the FTP spool directory.

As you can see, there are few things you can do to prevent break-ins. As more control you have over your system, more will be your alternatives to prevent it and even to try to catch an intruder... as far as system's at least. Legally speaking, unfortunately there is not much you can do yet. The legal system in U.S. is trying to move fast, but not fast enough, one may say.

## Persecuting the Hacker: What the Legal System has to Say

Computer security law is a new field, not yet established in the realms of law. The meaning of most technical computer terms are still a bit foreign or unclear in the courtrooms.

The legal establishment had yet to reach broad agreement on many key issues. Even the meaning of such basic terms as "data" can be the subject of contention.

Computer security law is still moving very slow, and if it moves, it is mostly due to litigation coming to court and making attorneys and judges very much reluctant due to their lack of knowledge and understanding of technical terms and security issues.

But the American Bar Association has already a full plate as computer security law and public policy needs to be developed. Needless to say, the legal perspective in pursuing a hacker is not in a solid ground yet, but the government and ABA acknowledges it and are working to resolve the issue.

The American Bar Association's (ABA) Science and Technology Section is responding to the control, legal and security issues associated with the Electronic Data Interchange (EDI) and the electronic commerce information technologies.

The Section has specialty committees in several areas under the Electronic Commerce and Information Technology Division:

- **The CyberNotary Committee** - The committee tries to address, and recommend solutions, to the discrepancies between international law and the U.S. law, which many times turns out to be inadequate to practice due to legal system's differences, costs and liabilities.
- The advent of electronic commerce demands a more reliable authentication and certification system of electronic "documents" to assure the reliability and enforceability of underlying acts, especially overseas.
- Although the legal and technical infrastructure doesn't yet exist, the CyberNotary Project has a proposal to rectify the lack of security in the international legal transactions as well as those that are placed electronically. The CyberNotary office is a concrete initiative that aims to bring together the information technology and the legal expertise.
- **The Information Security Committee** - The committee explores the computer security issues, included but not limited to those related to cryptograph, risk analysis, standards and commercial reasonableness and the relationship between security and the legal efficacy of electronic commerce.

- **Electronic Commerce Payment Committee** - This committee is dedicated to explore, considers the requirements of, and recommend on the legal solutions to meet the needs arising from undertaking electronic payments within the context of electronic commerce.
- **Judicial Electronic Data Interchange (EDI) Committee** - This Committee considers the use of EDI as the vehicle for administration of justice, among information systems of two or more parties.

**Note:**

If you would like to have more information about ABA's Science and Technology Section (STS) and its Committees, you can check the URL: [http:// http://www.intermarket.com/ecl/](http://http://www.intermarket.com/ecl/)

## ***What The Legal System Has To Say***

Back in 1990 the government began a nationwide campaign to crackdown on illicit computer hackers. There were several arrests, criminal charges, and even a dramatic show-trial, with several guilty pleas, and confiscation of data and equipment all over the country.

The U.S. Secret Service joined forces with state and local law enforcement groups throughout the nation to try to put a stop on the "computer underground" community, the hackers and crackers community. It was a showdown! There is even a book reviving those moments, *The Hacker Crackdown*, by Bruce Sterling ([bruces@well.sf.ca.us](mailto:bruces@well.sf.ca.us)).

**Tip:**

If you want to know more about this crackdown, check Bruce's electronic version of the book at URL: <http://homepage.eznet.net/~frac/crack.html>

The FBI's National Computer Crime Squad is dedicated to detecting and preventing all types of computer-related crimes. When an incident is detected, the tendency is to overreact, and many times, based on the legal infrastructure do deal with the issue, that is what ends up happening.

Network intrusions, for instance, have been made illegal by the U.S. federal government, but detection and enforcement are very difficult. The law, as it presents when facing computer crimes, is very much limited in essence and scope. It does not take much to realize it when you take the criminal case of Kevin Mitnick's, aka the Condor, when recently pleading bargain. His final plea and the crimes he allegedly committed had very little connection to each other.

**Note:**

If you are not familiar with Kevin Mitnick's case, he was arrested back in February of 1995 for allegedly breaking into the home computer of Tsutomu Shimomura, a respected member of the computer security world.

Kevin, also known as "Condor," was suspected of spoofing Tsutomu's computer and stealing computer security tools to distribute over the Internet. By the beginning of July, the federal prosecutors and Kevin's lawyers had reached a plea bargain agreement whereby Kevin would admit the charges of "possessing unauthorized access devices" in exchange of the prosecutors dropping 22 charges brought against him.

According to the sentence guidelines, Kevin's admitting he was guilty would carry a maximum prison sentence of eight months.

The fact is that corporations and governments alike, love to spy on the enemy. The Web is providing new opportunities for this. Wired Magazine (June of 1996), commented that more and more the American people are watching less television at night to spend an average of 11 hours in front of a computer screen. Mostly likely on the Web!

Hackers-for-hire became a trend. Somewhat, an idol. I had the chance, while writing this book, to talk to some of them, and what I found out is that there is a status quo in being a hacker. Just check the magazine and newspaper articles about them, it is always catching, vibrating. Crackers are living to become hackers, and hackers and law enforcement agents are re-living the tale (tale?).

Tracing hackers and crackers is a very labor-intensive, specially the first. Convictions are hard to be reached, as the laws are not written with electronic theft in mind.

For instance, how would you qualify a scenario where your site is victimized with mail bombing? Hackers, with little effort, can instruct a computer to repeatedly send electronic mail to your Webmasters account to such an extent that it could generate a "denial of service" state and potentially shutdown your entire site. Is this action illegal? It may not be.

**Note:**

The journalist Joshua Quittner and Michelle Slatalla had their home computer targeted by hackers and flooded with mail bombs. Also, their phone lines were rerouted for a whole weekend.

The problem is that, there is not a concrete definition yet for the term "computer-related crime." What is the difference between illegal or deliberate abuses of the Internet and an annoying act. Unfortunately, one could look at e-mail bombing both ways.

Legal systems everywhere, and ABA/STS is an example, are very busy trying to find ways of dealing with crimes and criminals on the Internet. As it stands, there is no common sense on how hackers and other computer criminals are prosecuted. It varies from one jurisdiction to another. It is as cases like

Kevin Mitnick and Jake Baker unfolds that the world's legal system starts to react and be ready for this new cast of citizens.

Computer information systems present a whole slew of legal issues. For instance, your Web site can very well be used for dissemination of useful information, but it can also be used as an outlet for defamation, contrabanded materials, etc. How should this situation be treaded? In case of a computer crime at your site where users where affected, who is liable, you or the "hacker?" Is the crime his fault, since he was the author, or the Webmaster's, since he controls and provides access to the site?

## ***The Current Regulations***

There are multiple ways for regulating a wide variety of crimes, or potential crimes, for that matter. Therefore, the regulatory environment governing computer information systems is still somewhat confused.

The Federal Communications Commission (FCC), is responsible for regulating broadcasters and common carriers providing electronic data. However, FCC does not regulate computer information systems because it is considered to be an "enhanced" service.

What the legal system has to say? Not much at the moment. Pursuing a hacker through the legal system is a much harder and "almost impossible" task then tracking he/she down on your system after an attack. However, there are case laws and statutes in existence that deals with some specific aspects of computer information systems:

- **Defamation** - It can occur on a computer information system, or your Web site, for the matter, in a number of forms. You, as the Web administrator, must guard your site and users from it. Defamation can occurs in two forms, libel and slander. Computer technology and other forms of technology makes it hard to distinguish between an act that was libel or slander. Many courts are advocating the elimination of the distinction. Speech on the Internet, for instance, has more of the characteristics of libel than slander. However, written or printed words are considered more harmful than spoken words because they are deemed more premeditated and deliberate.

Another issue, can a person sue for defamation that occurred to a fictitious name (username) or a persona that appears on a computer? Because defamation involves speech, defamation raises serious First Amendment concerns. As it stands, liability may result if the act was libelous, and may not, if the defamation concerns public figures, public officials, or matters of public interest.

- **Fighting Words** - This is a kind of speech that is not given First Amendment protection." If one of your users insult another user using words that are, by their very utterance inflicting injury or tending to incite, then he/she may have committed a crime.

***of the U.S. Code reads:***

***Whoever, with intent to extort from any person, firm association, or corporation, any money or other thing of value, transmits in interstate or foreign commerce any communication containing any threat to kidnap any person or any threat to injure the person of another, shall be fined not more than \$5,000 or imprisoned not more than twenty years, or both." This section was recently applied to convict a college freshman who sent an E-mail message to President Clinton threatening that "One of these days, I'm going to come to Washington and blow your little head off. I have a bunch of guns, I can do it."***

- **Child Pornography** - This is another area that is regulated. Recent international investigations into illegal child-pornography distribution via computer network have resulted in search warrants being issued to U.S. Customs agents in at least 15 states.



- **Fraud and Abuse** - The fraud and abuse statute states that it is a crime to fraudulently access or abuse the access to a computer.

**Note:**

The Fraud and Abuse Statute states that:

"a) whoever

1. Knowingly accesses a computer without authorization or exceeds authorized access, and by means of such conduct obtains information that has been determined by the United States Government pursuant to an Executive order or statute to require protection against unauthorized disclosure for reasons of national defense or foreign relations, or any restricted data, as defined in paragraph y. of section 11 of the Atomic Energy Act of 1954, with the intent or reason to believe that such information so obtained is to be used to the injury of the United States, or to the advantage of any foreign nation;
2. Intentionally accesses a computer without authorization or exceeds authorized access, and thereby obtains information contained in a financial record of a financial institution, or of a card issuer as defined in section 1602(n) of title 15, or contained in a file of a consumer reporting agency on a consumer, as such terms are defined in the Fair Credit Reporting Act (15 U.S.C. 1681 et seq.);
3. Intentionally, without authorization to access any computer of a department or agency of the United States, accesses such a computer of that department or agency that is exclusively for the use of the Government of the United States or, in the case of a computer not exclusively for such use, is used by or for the Government of the United States and such conduct affects the use of the Government's operation of such computer,

## of this section.

4. Intentionally accesses a Federal interest computer without authorization, and by means of one or more instances of such conduct alters, damages, or destroys information in any such Federal interest computer, or prevents authorized use of any such computer or information, and thereby
  - . causes loss to one or more others of a value aggregating \$1,000 or more during any one year period;
  - B. modifies or impairs, or potentially modifies or impairs, the medical examination, medical diagnosis, medical treatment, or medical care of one or more individuals; or
6. knowingly and with intent to defraud traffics (as defined in section 1029) in any password or similar information through which a computer may be accessed without authorization, if
  - (A) such trafficking affects interstate or foreign commerce; or
  - (B) such computer is used by or for the Government of the United States;

of this section.

*of this section is*

**(1) of this section which does not occur after a conviction for another offense under such subsection, or an attempt to commit an offense punishable under this subparagraph; and**

**(1) of this section which occurs after a conviction for another offense under such subsection, or an attempt to commit an offense punishable under this subparagraph; and**

**(1) of this section which does not occur after a conviction for another offense under such subsection, or an attempt to commit an offense punishable under this subparagraph; and (B) a fine under this title or imprisonment for not more than ten**

**years, or both, in the case of an offense under subsection**

**(6) of this section which occurs after a conviction for another offense under such subsection, or an attempt to commit an offense punishable under this subparagraph; and**

**(5) of this section which does not occur after a conviction for another offense under such subsection, or an attempt to commit an offense punishable under this subparagraph; and**

**(5) of this section which occurs after a conviction for another offense under such subsection, or an attempt to commit an offense punishable under this subparagraph.**

***The United States Secret Service shall, in addition to any other agency having such***

***authority, have the authority to investigate offenses under this section. Such authority of the United States Secret Service shall be exercised in accordance with an agreement which shall be entered into by the Secretary of the Treasury and the Attorney General.***

## Protecting Your Corporate Site

Unfortunately, despite the government, ABA and other agencies efforts, there is not much the law can do for you and your users. At least, not in a short term or without lots of time and money to battle in the legal system.

Your best bet then is to make sure you can protect your Web site. This whole book discussed the several ways you can protect your site, emphasizing on the use of firewalls. I hope you take to heart what was discussed and start putting in practice. It will save you time and grief.

Nevertheless, I would like to close this book by reminding you of the very basic Web site security requirements to keep in mind:

- Confidentiality
- Integrity, and
- Availability.

The only way you will be able to achieve excellence in these areas is going to be by regulating the flow of users, services and activity of your site with predetermined set of rules, the security policy.

The security policy, which should precede any of the security strategies you may implement (firewalls, packet filtering, access control and authentication, etc.), will specify which subjects can access which objects. Thus, it is important that you have very clear what are the subjects you will be dealing with at your site (e.g. internal users, external users, clients leasing portions of your site, etc.) and objects to be accessed or offered (e.g. Web services, links, leased home pages, etc.)

You will need to devote some time and effort to elaborate your security policy. Security measures are employed to prevent illicit tampering with your "users and clients as well as your services offered.

## Preventing Break-ins at Your Site

As your site gains momentum, your users will increasingly rely on the services you provide to carry out many essential functions of their day-to-day life. Luckily, your site will become part of their "bookmarks" right at the first visit. If your site is to be depended upon, it is essential that you,

Webmasters, systems administrators and everyone else responsible for its operation to recognize the vulnerabilities to which the site is subject and take the steps to implement appropriate safeguards. That is what this book is all about.

Internet security, while a relatively recent concern, is subject to a variety of interpretations. Historically, security measures have been applied to the protection of classified information from the threat of disclosure in a MIS or computer lab environment. But nowadays, in an environment where general Internet users are capable to shift from personal home pages on the Web to personal IPSs, much attention has been directed to the issue of individual privacy as it relates to personal information stored in the computerized data systems.

Data integrity in financial, scientific and process control applications at your site should be another focus of attention.

When setting up you security policy, remember that security policy, like your car insurance, is to a large extent applied risk management: you should try to achieve a tolerable level of risk at the lowest possible cost. The goal is to reduce the risk exposure of your site to an acceptable level, best achieved by a formal assessment of your risks. This includes a number of components, such as the identification of the Web site assets, values, threats and vulnerabilities, as discussed in this book, as well as the financial impact of each threat-asset combination.

When analyzing your risks, make sure to involve as many people as possible, from users to managers and upper management. If confidentiality is a specific concern, based on the services you will provide (dating services, financial services, etc.), additional protection must be provided through the application of hardware/software security solutions as well as mandatory regulatory requirements.

Make sure to include specific security administrative practices, assigning security responsibilities to all professionals involved with the operation and maintenance of the site. Make sure to determine:

- A procedure to ensure that risks are identified (auditing logs, printing activity reports, implementation of configuration and security checklists. Etc.);
- Individual security duties and the appropriate assignment of responsibilities;
- File access policy and designated restricted areas in your disk farm;
- Authorization and authentication procedures for new users and services (you can't be in control of the site 24 hours a day, seven days a week! Have a documented set of procedures);;
- A contingency plan, in case of emergencies.

## Final Considerations

Internet security is possible! Break-ins, although inevitable, can be prevented when you are aware of security issues. However, this means that all the issues discussed in this book would have to be taken into consideration, that this book would not be enough to guide you in securing you site as new threats arise everyday and the Web and computer technology is to diverse to be comprised in a single book.

Internet security and firewall management is a full-time job. The systems manager or the Webmaster, or the LAN administrator, or the MIS director and so on, cannot be blamed in case of an incident. They probably didn't have time!

If the people involved with the every-day activity and operation of your site could know the information contained in this book, the risks of your site being attacked and the resulting damage could be much less. This is exactly why I wrote this book. Not for the gurus. Not for the technically skilled. It was to give an survey of the threats a Web site faces, the existence of hackers to make our work more... enjoyable.

By no means, don't you have the illusion that by following the information and guidelines of this book your site will be impossible to penetrate.

Finally, make sure to keep security into account during the whole phase of your Web site design. This will also make security much more user friendly.

[Orders](#)[Orders](#)[Back](#)[Forward](#)[Comments](#)[Comments](#)

---

[COMPUTING MCGRAW-HILL](#) | [Beta Books](#) | [Contact Us](#) | [Order Information](#) | [Online Catalog](#)

---

HTML conversions by [Mega Space](#).

This page updated on December 05, 1997 by [Webmaster](#).

[Computing McGraw-Hill](#) is an imprint of the [McGraw-Hill Professional Book Group](#).

Copyright ©1997 [The McGraw-Hill Companies, Inc.](#) All Rights Reserved.

Any use is subject to the rules stated in the [Terms of Use](#).

[Orders](#)[Orders](#)[Back](#)[Forward](#)[Comments](#)[Comments](#)

© 1997 The McGraw-Hill Companies, Inc. All rights reserved.

Any use of this Beta Book is subject to the rules stated in the [Terms of Use](#).

---

# Chapter 11

## Proxy Servers

Application gateways, or proxy server, define a whole different concept in terms of firewalls. In order to balance out some of the weaknesses presented by packet-filtering routers, you can use certain software applications in your firewall to forward and filter connections for services, such as telnet and FTP. These applications are referred to as a proxy service, and the host running the proxy service is often called an application gateway.

Many IS&T (information systems and technology) professionals consider application gateways to be a true firewall because the other types lack user authentication. Accessibility is much more restricted than with packet-filtering and circuit-level gateways because it requires a gateway program for every application such as telnet, FTP, and so on.

As a matter of fact, there are many companies that only use a proxy service as they firewall, while others just rely on the firewall itself. Depending of your environment, size of your company and the level of protection you want to accomplish, one or the other may be all you need. However, as a rule of thumb, you should always consider the implementation of a proxy service combined with your package-filtering routers (firewalls), so that you can achieve a more robust level of defense and flexible access control. Also you will find that many firewall products will bring you the best of both worlds, combining both filtering and proxying features in a single package.

The combination of application gateways and packet-filtering routers to increase the level of security and flexibility of your firewall is therefore the ideal solution for addressing Internet security. These are often called hybrid gateways. They are somewhat common, as they provide internal hosts unobstructed access to untrusted networks while enforcing strong security on connections coming from outside the protected network.

Consider figure 11.1 as an example of a site that uses a packet-filtering router and blocks all incoming telnet and FTP connections. The router allows telnet and FTP packets to go only to the telnet/FTP application gateway. A user connecting to a site system would have to connect first to the application gateway, and then to the destination host, as follows:

1. A user telnets to the application gateway and enters the name of an internal host
2. The gateway checks the user's source IP address and accepts or rejects it according to any access

criteria in place

3. The user might need to be authenticated
4. The proxy service creates a telnet connection between the gateway and the internal server
5. The proxy service passes bytes between the two connections
6. The application gateway logs the connection

If you look at figure 11.2, it shows the details of the virtual connection happening on figure 11.1 and emphasizes the many benefits to using proxy services. Lets stop for a moment and try to identify some of these benefits:

Proxy services allow through only those services for which there is a proxy. If an application gateway contains proxies for FTP and telnet, only FTP and telnet are allowed into the protected subnet. All other services are completely blocked. This degree of security is important. Proxy makes sure that only trustable services are allowed through the firewall and prevents untrusted services from being implemented on the firewall without your knowledge.

Let's take a look at some advantages and disadvantages of application gateways.

There are several advantages to using application gateways over the default mode of permitting application traffic directly to internal hosts. Here it is the five main ones:

1. Hiding information. The names of internal systems (through DNS) are hidden to outside systems. Only the application gateway host name needs to be known to outside systems.
2. Robust authentication and logging. The traffic can be pre-authenticated before it reaches internal hosts. It can also be logged more efficiently than if logged with standard host logging.
3. Cost-effectiveness. Authentication/logging software and hardware are located at the application gateway only.
4. More comprehensive filtering rules. The rules at the packet-filtering router are more comprehensive than they would be with the routers filtering and directing traffic to several specific systems. With application gateways, the router needs only to allow application traffic destined for the application gateway and block the rest.
5. E-mail. It can centralize e-mail collection and distribution to internal hosts and users. All internal users would have e-mail addresses of the form user@mailbag, where mailbag is the name of the e-mail gateway. The gateway would receive mail from outside users and then forward it to internal systems.

However, nothing is perfect! Application gateways have disadvantages too. To connect to client-server protocols such as telnet requires two steps, inbound or outbound. Some even require client modification, which is not necessarily the case of a telnet application gateway, but it would still require a modification in user behavior. The user would have to connect to the firewall as opposed to connecting directly to the host. Of course, you could modify a telnet client to make the firewall transparent by allowing a user to specify the destination system (as opposed to the firewall) in the telnet command. The firewall would still serve as the route to the destination system, intercepting the connection and running authentication procedures such as querying for a one-time password.

You can also use application gateways for FTP, e-mail, X Window, and other services.



**Note:**

Some FTP application gateways have the capability to block put and get commands to specific hosts. They can filter the FTP protocol and block all put commands to the anonymous FTP server. This guarantees that nothing can be uploaded to the server.

So, what are proxies after all? Simply put, proxy are gateway applications basically used to route Internet and web access from within a firewall.

If you have used TIA (The Internet Adapter) or TERM, you probably are familiar with the concept of redirecting a connection. Using these programs, you can redirect a port. Proxy servers work in a similar way, by opening a socket on the server and allowing the connection to pass through.

A proxy is a special HTTP server that typically is run on a firewall. A proxy basically does the following:

- Receives a request from a client inside the firewall
- Sends this request to the remote server outside of the firewall
- Reads the response
- Sends it back to the client

Usually, the same proxy is used by all of the clients in a subnet. This enables the proxy to efficiently cache documents that are requested by several clients. Figure 11.3 demonstrates these basic functions.

The fact that a proxy service is not transparent to the user means that either the user or the client will have to be proxified. Either the user is instructed on how to manage the client in order to access certain services (telnet, FTP), or the client, such as Web clients, should be made proxy-aware.

The caching of documents makes proxies very attractive to those outside the firewall. Setting up a proxy server is not difficult. Today, most web client programs already have proxy support built in. It is very simple to configure an entire workgroup to use a caching proxy server, which helps to cut down on network traffic costs because many of the documents are retrieved from a local cache after the initial request has been made.

Proxy has a mechanism that makes a firewall safely permeable for users in an organization without creating a potential security hole through which hackers can get into the organization's protected network.

This application-level proxying is easily supported with minor modifications for the Web client. Most standard out-of-the-box Web clients can be configured to be a proxy client without any need for compilations or special versions. In a way, you should begin to see proxying as a standard method for getting through firewalls, rather than having clients getting customized to support a special firewall method. This is especially important for your Web clients because the source code will probably not be available for modification.

As an example of this procedure, check the Anonymizer site, at URL <http://www.anonymizer.com>. All connections passing through the Anonymizer are proxified. The output connection was totally redirected and had its address changed, only that here it is done to protect the identity of the client, rather than

access control (another benefit of using proxies!). Clients without DNS (Domain Name Service) can still use the Web because the only thing they need is proxy IP addresses.

**Tip:**

You can build a proxy-type firewall by using TIS toolkit if you have experience with UNIX and programming. It contains proxies for telnet, FTP, Gopher, Rlogin and a few other programs. Also, as an alternative, you can use Purveyor 1.1 (<http://www.process.com>), which offers all of that without a need for UNIX and programming knowledge. Best of all, you won't need an expensive UNIX box, it runs on Windows NT and Windows 95.

Organizations using private network address spaces can still use your Web site as long as the proxy is visible to both the private internal net and the Internet, most likely using two separate network interfaces.

Proxying permits high-level logging of client transactions, which includes the client IP address, date and time, URL, byte count, and success code. Another characteristic of proxying is its capability to filter client transactions at the application-protocol level. It can control access to services for individual methods, server and domain, and so on.

As far as caching, the application-level proxy facilitates it by enabling it to be more effective on the proxy server than on each client. This helps to save disk space because only a single copy is cached. It also enables more efficient caching of documents. Cache can use predictive algorithms such as look ahead and others more effectively because it has many more clients with a much larger sample size on which to base its statistics.

Have you ever thought about browsing a Web site when the server is down? It is possible, if you are caching. As long as you connect to the cache server, you can still browse the site even if the server is down.

Usually, Web clients' developers have no reason to use firewall versions of their code. But in the case of the application-level proxy, the developers might have an incentive: caching! I believe developers should always use their own products, but they usually don't with firewall solutions such as SOCKS. Moreover, you will see that a proxy is simpler to configure than SOCKS, and it works across all platforms, not only UNIX.

Technically speaking, as shown in figure 11.4, when a client requests a normal HTTP document, the HTTP server gets only the path and keyword portion of the requested URL. It knows its hostname and that its protocol specifier is http:.

When a proxy server receives a request from a client, HTTP is always used for transactions with the proxy server, even when accessing a resource served by a remote server using another protocol such as Gopher or FTP.

A proxy server always has the information necessary to make an actual request to remote hosts specified in the request URL. Instead of specifying only the pathname and possibly search keywords to the proxy server, as figure 11.5 shows, the full URL is specified.

This way, a proxy server behaves like a client to retrieve a document, calling the same protocol module of Libwww that the client would call to perform the retrieval. However, it is necessary to create an HTTP containing the requested document to the client. A Gopher or FTP directory listing is returned to the client as an HTML document.

**Caution:**

Netscape does not use libwww so if you are using Netscape, you would not be calling a protocol module of libwww from the client.

Therefore, by nature a proxy server has a hybrid function: It must act as both client and server. A server when accepting HTTP requests from clients connecting to it, and a client (to the remote) to actually retrieve the documents for its own client.

**Note:**

In order for you to have a complete proxy server, it must speak all of the Web protocols, especially HTTP, FTP, Gopher, WAIS, and NNTP.

One of the HTTP server programs, CERN's httpd, has a unique architecture. It is built on top of the WWW Common Library. The CERN httpd speaks all of the Web protocols just like Web clients, unlike other HTTP servers built on the WWW Common Library. It has been able to run as a protocol gateway since version 2.00, but not enough to act as a full proxy. With version 2.15, it began to accept full URLs, enabling a proxy to understand which protocol to use when interacting with the target host.

Another important feature with a proxy involving FTP is that if you want to deny incoming connections above port 1023, you can do so by using passive mode (PASV), which is supported.

**Caution:**

Not all FTP servers support PASV, causing a fallback to normal (PORT) mode. It will fail if incoming connections are refused, but this is what would happen in any case, even if a separate FTP tool were used.

However, before considering caching, you should be aware of at least couple problems that can occur and need to be resolved:

- Can you keep a document in the cache and still be sure that it is up-to-date?
- Can you decide which documents are worth caching, and for how long?

The caching mechanism is disk-based and persistent. It survives restarts of the proxy process as well as restarts of the server machine itself. When the caching proxy server and a Web client are on the same machine, new possibilities are available. You can configure a proxy to use a local cache, making it

possible to give demos without an Internet connection.

A great feature of the HTTP protocol is that it contains a HEAD method for retrieving document header information without having to retrieve the document itself. This is useful to tell you if the document has been modified since your last access. But in cases where the document has changed, you have to make a second connection to the remote server to do the actual GET command request to retrieve the document. Therefore the HTTP protocol needs to be extended to contain an If-modified-Since request header, allowing it to do a conditional GET request.

In case the document has not been modified since the date and time specified, a 304 (Not modified) response will be returned along with a special result code. If the document has been modified, the reply will be as if the request was just a normal GET request.

**Tip:**

All major HTTP servers already support the conditional GET header.

Just for your information, there is a function called no-cache pragma, which is typically used by a client's reload operation. This function provides users with the opportunity to do a cache refresh with no visible modifications in the user interface. A no-cache pragma function is forwarded by the proxy server, thus ensuring that if another proxy is also used, the cache on that server is ignored.

In summary, taken from the internal network perspective, a proxy server tends to allow much more outbound access than inbound. Generally, it will not allow Archie connections or direct mailing to the internal network, you will have to configure it.

Also, depending on which proxy server you are using, you should anticipate problems with FTP when doing a GET or an ls because FTP will open a socket on the client and send the information through it. Some proxy server will not allow it, so if you will be using FTP, make sure the proxy server supports it.

**Note:**

With Purveyor, a client who does not implement Domain Name Services (DNS) will still be able to access your Web site through Purveyor's proxy server. The proxy IP address is the only information required.

As the applications for proxies rise, there are many features that are still in their early stages, but the basic features are already there! You should plan on having a proxy server on your firewall. Although caching is a wide and complicated area, it is also one of the parts of the proxy server that needs to be improved.

**Tip:**

You can provide Internet access for companies using one or more private network address spaces, such as a class A IP address 10.\*.\* by installing a proxy server that is visible to the Internet and to the private network.

I believe the HTTP protocol will be further enhanced as Internet growth continues to explode. In the near future you should see multipart requests and responses becoming a standard, enabling both caching and mirroring software to refresh large amounts of files in a single connection. They are already much needed by Web clients to retrieve all of the inlined images with one connection.

Moreover, proxy architecture needs to be standardized. Proxy servers should have a port number assigned by Internet Assigned Numbers Authority (IANA). On the client side, there is a need for a fallback mechanism for proxies so that a client can connect to a second or third proxy server if the primary proxy failed (like DNS). But these are just items on a wish list that will certainly improve netsurfing but are not yet available.

**Tip:**

If you need to request parameter assignments (protocols, ports, etc) to IANA, they request you to send it by mail to [iana@isi.edu](mailto:iana@isi.edu). For SNMP network management private enterprise number assignments, please send e-mail to [iana-mib@isi.edu](mailto:iana-mib@isi.edu).

Taking into consideration the fast growth of the Web, (by the time I finish this chapter, the Web will have surpassed FTP, and gopher all together!), , I believe proxy caching represents a potential (and needed)." Bits and bytes will need to get returned from a nearby cache rather than from a faraway server in a geographically distant place.

## SOCKS

SOCKS is a packet that enables servers behind the firewall to gain full access to the Internet. It redirects requests aimed at Internet sites to a server, which in turn authorizes the connections and transfers data back and forth.

**Tip:**

If you need more information about SOCKS, you can find it at <http://www.socks.nec.com>. To join the SOCKS mailing list, send mail to [majordomo@syl.dl.nec.com](mailto:majordomo@syl.dl.nec.com) with subscribe SOCKS your@e-mail.address in the body of the mail.

SOCKS was designed to allow servers behind a firewall to gain full access to the Internet without requiring direct IP reachability . The application client establishes communication with the application

server through SOCKS. Usually the application client makes a request to SOCKS, which typically includes the address of the application server, the type of connection, and the user's identity.

After SOCKS receives the request, it sets up a proper communication channel to the application server. A proxy circuit is then established and SOCKS, representing the application client, relays the application data between the application client and the application server.

It is SOCKS that performs several functions such as authentication, message security-level negotiation, authorizations, and so on while a proxy circuit is being set up.

SOCKS performs four basic operations (the fourth being a feature of SOCKS V5):

- Connection request
- Proxy circuit setup
- Application data relay
- Authentication (V5)

Figure 11.6 shows a control flow model of SOCKS.

Authentication methods are decided by SOCKS based on the security policy clauses that it defines. If none of the methods declared by the client meets the security requirement, SOCKS drops the communication.

As depicted on figure 11.7, after the authentication method is decided upon, the client and SOCKS begin the authentication process using the chosen method. In this case, SOCKS functions as a firewall.

Through an authentication procedure called GSS-API (Generic Security Service Application Program Interface), clients negotiate with SOCKS about the security of messages. Integrity and privacy are the options that can be applied to the rest of messages, including the proxy requests coming from the application client as well as Socks' replies to the requests and its application data.

As far as UDP-based applications, SOCKS V5 has a connection request: the UDP association. It provides a virtual proxy circuit for seamlessly traversing UDP-based application data. However, be careful here! The proxy circuit for TCP-based applications and UDP-based ones are not the same. They mainly differ in two ways:

- UDP's proxy circuit, a pair of address information of the communication end-points, necessary for sending and receiving datagrams.
- Application data, which is encapsulated by UDP proxy headers that include, along with other information, the destination address of a given datagram.

You can use SOCKS in different network environments. Figure 11.8 shows an example of one of the most popular setups.

A single SOCKS can be utilized as a firewall. SOCKS V5 supports authenticated traversal of multiple firewalls, extending it to build a virtual private network as shown in the figure.

The great advantage of the existing authentication scheme integrated into SOCKS is that the centralized network access of SOCKS enables the enforcement of security policy and the control of network access much easier than without centralized access. You need to watch for the fact that these access points

unfortunately can become the bottleneck of internetworking. You must try to balance it out with the hierarchical distribution of SOCKS, shadow SOCKS (multiple parallel SOCKS), and other mechanisms for keeping the consistency of your security policy. Also, beware of potential security holes and attacks among multiple SOCKS, and so on, as a factor of acceptability of SOCKS as a secure mechanism for insecure network.

The integration of SOCKS and the Web has substantially increased the area of security on the web. Whereas secure web-related technologies such as S-HTTP (Security-enhanced HyperText Transport Protocol) and SSL (Secure Socket Layer) provide message and server authentications, SOCKS can be successfully integrated to provide user authentication and authorization. Furthermore, the security technologies employed on the Web can also be integrated into SOCKS to enhance the security of proxy connections.

## Tcpd, the TCP Wrapper

You should be aware that the TCP Wrapper is not really a firewall utility but provides many of the same effects. By using TCP Wrapper, you can control who has access to your machine and to what services they have access to. It also keeps logs of the connections and does basic forgery detection.

TCP Wrapper was written by Wietse Venema of The Netherlands' Eindhoven University of Technology. The key source of it is `tcpd`, a simple wrapper that in action envelopes every network daemon run by `inetd`. The `tcpd` wrapper is a simple, great tool to write rules based on acceptance or denial of connections. It also enables you to finger a host that attempts to illegally request an `rlogin`, for example.

You can use `tcpd` as an auditing tool. It has the capability to log attempted network connections to the wrapper service, which can greatly improve security. Although it has great features, in order for you to use it, you have to be connected to the Internet thus requiring an IP address.

### Tip:

If you want to take a look at the source code for TCP Wrapper, you can download it from <ftp://ftp.win.tue.nl/pub/security>.

Another feature of TCP Wrapper is its support library, `libwrap.a`. It can be used by many other programs to provide the same wrapper-like defenses of other services.

Also, it only controls the machine it is installed on, making it a poor choice for network use. Firewalls are much more broad and therefore can protect every machine of every architecture.

However, the major drawback of TCP Wrapper is that it does not work on Apple Macintoshes or Microsoft Windows machines. It's basically a UNIX security tool.

# Setting Up and Configuring the Proxy Server

In order to set up my proxy server I need additional software. For this situation, I need SOCKS.

**Note:**

You can download SOCKS from

`ftp://sunsite.unc.edu/pub/Linux/system/Network/misc/socks-linux-src.tgz`

If you care to, you can also download a configuration example, found in the same directory, called `socks-config`.

By the time I start configuring SOCKS, I should be aware that SOCKS needs two separate configuration files: one to notify the allowed access and the other to route the requests to the appropriate proxy server. I have to make sure the access file is loaded on the server and that the routing file is loaded on every UNIX computer.

I will be using SOCKS version 4.2 beta, but as discussed earlier in this chapter, version 5 is already available. If you're also using version 4.2 beta, the access file is called `sockd.conf`. Simply put, it should contain two lines: a permit line and a deny line. For each line I will have three entries:

- The identifier (permit/deny). It will be either permit or deny, but I must have both a "permit" and a "deny" line.
- The IP address. It holds up to four byte address in typical IP dot notation.
- The address modifier. A typical IP address four byte number, acting like an netmask, such as 255.255.255.255.

For example, the line will look like this:

```
permit 192.168.2.26 255.255.255.255
```

My goal is to permit every address I want and then deny everything else. Another issue I have to decide is about power users or special ones. I could probably allow some users to access certain services, as well as deny certain users from accessing some of the services that I have allowed in my internal network.

However, this is done by using `ident`, an application that if on, will have `httpd` connect to the `ident` daemon of the remote host and find out the remote login name of the owner of the client socket. Unfortunately the Trumpet Winsock I am using does not support it, nor do some other systems. Keep in mind that if your system supports `ident`, this is a good feature to use, even though it's no trustworthy, you should use it for informational purpose only, as it does not add any security to your system.

One thing I need to watch out for, and I am sure you will have to as well, is not to confuse the name of the routing file in SOCKS, `socks-conf`, with the name of the access file. They are so similar that I find it easy to confuse the two. However, their functions are very different.



The routing file is there to tell SOCKS clients when to use it and when not to use it. Every time an address has a direct connection to another (through Ethernet, for example), SOCKS is not used because its loopback is defined automatically. Therefore, I have three options here:

- To deny, which tells SOCKS to reject a request.
- To direct, which tells us what address should not use SOCKS (addresses that can be reached without SOCKS).
- To sockd, which tells the computer what host has the SOCKS server daemon on it (the syntax is sockd @=<serverlist> <IP address> <modifier>). The @= entry enables me to enter a list of proxy servers IP addresses.

Now, to have my applications working with the proxy server, they need to be "sockified." I need a telnet address for direct communication and another for communications using the proxy server. The instructions to sockify a program are included with SOCKS. Because the programs will be sockified, I will need to change their names. For example, finger will become finger.orig, ftp will become ftp.orig, and so on. The include/socks.h file will hold all of this information.

A nice feature of using Netscape Navigator is that it handles routing and sockifying itself. However, there is another product I plan to use called Purveyor Web Server 1.2. that not only also works as a proxy for FTP, Gopher, and HTTP but also

But one of the reasons I will be using Trumpet Winsock (for Microsoft Windows) is that it comes with built-in proxy server capabilities. I just need to enter the IP address of the server and addresses of all the computers I can reach directly in the setup menu. Trumpet Winsock will then handle all of the outgoing packets.

At this point, I should be done. However, I know I'll have a problem (and you will too!). SOCKS does not work with UDP, only with TCP. Programs such as Archie use UDP, which means that because SOCKS is my proxy server, it will not be able to work with Archie. Tom Fitzgerald (fitz@wang.com) designed a package called UDPrelay to be used with UDP, but it's not compatible with Linux yet.

|          |          |
|----------|----------|
| Orders   | Orders   |
| Comments | Comments |

|      |         |
|------|---------|
| Back | Forward |
|------|---------|

---

[COMPUTING MCGRAW-HILL](#) | [Beta Books](#) | [Contact Us](#) | [Order Information](#) | [Online Catalog](#)

---



HTML conversions by [Mega Space](#).

This page updated on December 05, 1997 by [Webmaster](#).

[Computing McGraw-Hill](#) is an imprint of the [McGraw-Hill Professional Book Group](#).

Copyright ©1997 [The McGraw-Hill Companies, Inc.](#) All Rights Reserved.

Any use is subject to the rules stated in the [Terms of Use](#).

|        |        |
|--------|--------|
| Orders | Orders |
|--------|--------|

|      |         |
|------|---------|
| Back | Forward |
|------|---------|

|          |          |
|----------|----------|
| Comments | Comments |
|----------|----------|

© 1997 The McGraw-Hill Companies, Inc. All rights reserved.  
Any use of this Beta Book is subject to the rules stated in the [Terms of Use](#).

---

# Chapter 12

## *Firewall Maintenance*

The level of security you have implemented at your company is directly related to the amount of money you invested on it and the risks you're willing to take. So you install a firewall!

Firewall maintenance begins with its management, and as part of management you must not consider the installation of a firewall as the solution to all of your security problems. Always keep in mind, as stressed throughout this book, that firewalls provide a wide variety of controls, but in the end they are only a tool. A firewall is part of a diversified defense strategy that identifies what must be protected and identifies the potential threats.

It seems obvious, but there is more to protecting a network than hardware and software. Security comes from the integration of reliable technology, active and alert systems administrators, and management decisions regarding user access to the Internet and other computer resources. Prudence demands the development of a comprehensive plan to deal with system security. You, as the administrator, along with your security staff, will have to define at least:

1. Which assets to be protected, and
2. What is the level of risks those assets are exposed to.

Therefore, your security policy must include multiple strategies. Increasingly this is overlooked as administrators turn toward technology and firewalls in particular (and rely on them!), as a cure-all for their installation's security. This is a dangerous path to follow. Firewalls should not be called upon to perform increasingly complex and unreasonable tasks such as scanning packets for viruses, encrypted data and even foreign languages.

Now, the firewall should not be forgotten either. It's not because it's doing its job that you just we'll let it alone. Just like a car, in order for it to run well and efficiently it will require continuous care and attention. Some times, an occasional drill will be necessary, as well as few checkups. Never neglect a firewall! The Internet is a wild thing! If today your firewall is set up to protect your corporation from the known threats out there, tomorrow there might be a new one you're not aware of and it will come to bite you.

How much time you will have to allocate to care for your firewall will vary. It will depend upon the type

of firewall you have installed, the assets you're protecting and the kind of Internet services and access you're providing.

Some company's rely on routers to filter unwanted traffic connection. If that is your case, what you have is a set of rules not so complicated to maintain. As discussed before, with this kind of firewall, you're either allowing or denying connections. In this case, I have a good and a bad news for you: the good, the amount of time you will need to spend in caring for your firewall is almost none. Except for allowing new connections or denying some more, there's nothing more you can do, other than make sure that firewall is on and the NIC cards are still alive, which in case of failure you will notice right away anyway. The bad news is that you may be preventing desired traffic to come in, such as potential new customers, and not taking advantage of lots of Internet services and resources at Cyberspace. Make sure not to develop a bad rep for MIS!

If you are one of the Fortune 500 company, you better have a complete and detailed security policy. Otherwise you may be in for a ride! At the very least, you should be probing the network traffic coming to the firewall from the Internet, as well as leaving your protected network daily. Don't be surprise if your traffic measuring hits the gigabytes! Thus, to perform these probing manually is literally impossible. Your firewall must offer traffic probing, security alerts and report generation features.

Since firewalls are usually in an ideal position to gather usage statistics, as all traffic must pass through them, you will be able to track usage of the network link on regular intervals and analyze them. These analysis can greatly help you assess network usage and performance, as well as any security threat and countermeasure.

For instance, you can analyze which protocols are delivering the best performance, which subnets are the most accessed, and even, based on the information you collect, schedule service upgrades, bug fixes or if necessary, discover a security hole and plug it.

If you have a packet filter firewall, you should have at least a basic understanding of the transport protocols that they see crossing the wires so you can care for it. In doing so, the filter rules that you will likely use, Alec Muffett well outlines in his paper (Alec.Muffett@UK.Sun.COM) are typically to control traffic on the basis of:

- **Transport endpoints**, or a notion of what's inside and what's outside the network. In the TCP/IP world, this is usually implemented by masking off portions of the source and destination addresses, and checking whether or not the remaining parts of the addresses refer to hosts inside the secured network.
- **Transport protocol**, such as TCP, UDP, or raw IP. Other protocols may or may not be directly supported, or it may be assumed that they are to be tunneled through the firewall.
- **Protocol options**, should be featured in any good firewall, which also should have the ability to "drop" traffic on the basis of protocol-dependent options which might compromise security if misused - for instance, the IP "source routing" option which can be utilized in traffic forgery.

Muffett indicates that similar issues arise when trying to ensure proper handling of ICMP packets, for example, in trying to control messages necessary for the proper operation of IP. Further, he alerts that the most critical facility in a packet filter is the ability to match network traffic against a table of permitted source and destination hosts (or networks), but it is also vitally important to note that the firewall's checking must be done against both ends of a connection, and must take into account the service port

numbers at each end of the connection, otherwise the firewall may be trivially subverted.

# Keeping Your Firewall in Tune

Tuning a firewall is just like bring your car for a tune up. As with your car, a tune up of a firewall is necessary because you will,

- Extend its life,
- Certify that it is running properly,
- Ensure the firewall is still promoting the secure environment to your corporation, as it was designed and implemented for,
- Optimize its operation and services,
- Perform any necessary upgrades, and
- Make sure all the firewall components are still functioning and interacting with each other.

By periodically performing a tune-up in your firewall you will be able to fairly evaluate the load your firewall is taking and/or is capable of bearing and anticipate future problems or issues. By modeling its performance against a scaled number of measured loads you will be able to have a good picture of your firewall vital signs.

The following section was based on a great paper written by Marcus J. Ranum (mjr@nfr.net), CEO of Network Flight Recorder, Inc.

## Note:

For more information of firewall, check Marcus Ranum's personal Web site at <http://www.clark.net/pub/mjr> or <http://www.nfr.net>.

The following is a firewall tune-up procedure I recommend you to do so you can have a "health chart" of your system:

1. Monitor your firewall for a month and store all the log results. AS many logs you have more accurate and complete will be results of your firewall "physical" exam!

By doing this you will have a first hand idea of the load going through your firewall, regardless if it is a packet or application level firewall. If the firewall is an application level firewall, this should be a breeze, as these firewall already provide you a lot of reports about the system by default. Now, if it is a packet-level firewall, like a router for example, then you will have to develop some kind of log-reduction, by using tools such as tcpdump, see figure 12.2 or NNstat.

2. Sort the logs by the time of day, per hour.

Notice that some hours of the day have higher peaks than others and exhibit different load characteristics. Sorting the logs at an hour interval evidences it.

3. Tabulate the batch of logs by services, yielding values like:
  - Number of email messages during that interval

- Average size of email messages during that interval
- Average time between arrival of email during that interval
- Number of web hits during that interval
- Average size of retrieved web objects during that interval
- Average time between web accesses during that interval
- Number of FTP retrieves during that interval
- Average size of FTP objects retrieved during that interval
- Average time between FTP retrievals during that interval
- Number of TELNET sessions during that interval
- Maximum concurrent TELNET sessions during that interval
- Amount of NNTP traffic in during that interval
- Amount of NNTP traffic out during that interval
- Average time between NNTP sessions during that interval

1. Note the peak load in any one interval for each service wherever it occurs.

If you were to put all that load through the firewall in one interval's worth of time, you would have a clear picture of the worst-case load you have yet observed.

2. Implementation tools to generate these loads from existing logs would be pretty straightforward and could be run at any site wishing to perform this test. Presumably the values would be different for each site but maybe not very. A number of expect scripts or PERL scripts, using static file data could simulate the load through the firewall without having to actually do the work.
3. After this procedure you should have a basic "workload by interval" paradigm for your firewall in your company's environment, including peaks and a worst case scenario. With this data on hand you will be able to tune up your firewall based on the assumption of "whatif" the rate of load increase, by watching what happens to the rate of service requests between busy hours and non-busy hours.

Notice that you can could on the "workload by interval" result as sustainable by your firewall because you measured it, right? The goal here is to find out how far your firewall can go, from the doable load level and the load level at which the firewall topples.

4. Now you can write a test harness that invokes the emulators in a way that will develop the same load model. Values that you should control are:
  - Number of concurrent loads for service X
  - Size of accesses for service X
  - Interval between accesses for service X
1. Run the test harness with the load configured to match the loadout at a given time that is not peak but someplace near it.
2. Compare the run times for the near peak load with the actual measured near peak load.

Notice that hey should be about the same!!

3. Run the test harness to emulate the peak load

4. Now compare the run times for the peak load with the run times for the actual measured peak load.

Again, notice that they should be about the same!!

5. Now, you have a template to fine tune your firewall, based on what happens to it when the traffic load increases above real measured values!!

## Monitoring Your System

A firewall configuration must have a management module, pay close attention to this when selecting your firewall product. Chapter 14, "Types of Firewalls," provides information about the management features of the main firewalls available on the market. Management features are key for monitoring your firewall and inspecting its functioning.

When monitoring your system, you should be concern about protecting the confidentiality of your users, the sensitivity of your devices and the security of your network in general. Most firewall do feature inspection mechanisms, many provide authentication and encryption.

A good precaution when preventing attacks and unnecessary risks on your firewall is to strip your bastion host down to the barest minimum of required functionality. I recommend you to remove all unnecessary software. Why would you risk running possibly dangerous software on the most critical host on your network? Why have the software installed on that machine at all?

Change the shells that are associated with passive systems accounts to something harmless like `/bin/false` - or better still, install a custom "shell" of some sort that triggers an alarm when invoked, and use that.

You should monitor password usage. There are several tools out there to check the security of passwords used by your users, just stop by at URL <http://www.simtel.net/nt> (if you have an Windows NT-based system) and check the security section. Consider using those digital token authentication devices.

## *Monitoring the Unmonitored Threats*

A firewall is not a final solution. You will need more then just a firewall to really secure your site. Thus, a firewall may not be able to tell you everything that's going on in your gateway, who's leaving, who's coming, especially "what's" coming! For instance, most firewalls could not protect against an E-Mail message destined for delivery to a valid user. So what for e-mail messages with attachments! These attachments could turn out to be a Trojan horse, a malicious applet, that could turn off your firewall from inside or bomb your protected network! So never consider the installation of a firewall a complete solution to all of your security woes.

Firewalls can screen traffic, and do so very effectively, but they do not give total protection from the contents of the data therein. A complete solution requires you to undertake security measures at all levels of network usage, from application access (access control and encryption) through the network layer (preventing spoofing, etc.) to the physical layer (restricting unauthorized connections to your network). That's why I started this book talking about the main security issues with the Internet services, on chapter 1, "Internetworking Protocols and Standards: An Overview," and 3, "Cryptography: Is it Enough?" I suggest you to read those chapters so that you can better understand the weaknesses of these services and

help your firewall to promote security by plugging the holes of the services you must need to use and making sure the information assets of your company (e-mail messages and documents, including financial data) are protected. By doing this, soon you will conclude that network security is not just a job for a firewall.

### Tip:

You should subscribe to GreatCircles firewall maillist archive at URL <ftp://ftp.greatcircle.com/pub/firewalls/>. This list has plenty reference material, and past issues of the digested "Firewalls" maillist. To subscribe send message to [majordomo@greatcircle.com](mailto:majordomo@greatcircle.com) and write in the body of the message: `subscribe firewalls you@your.email.addro`.

Also, check the TIS Firewall Toolkit at URL <ftp://ftp.tis.com/pub/firewalls/>. You will find here a free set of proxy clients and associated firewalling tools, as well as many technical papers upon the subject.

## Preventive and Curative Maintenance

In order to keep your firewall in a good shape, you must provide some maintenance to it, and in doing so, it is very important that you keep talking to your vendors. Watch for reports of security patches; ask your vendors about it. Participate on firewall lists, such as the one hosted by Brent Chapman, of GreatCircles, which I strongly recommend (<http://www.greatcircles.com>). Watch for new patch releases for your firewall of operating system, and when they come out, apply them. But wait! Make sure to confirm with your vendor that this new patch is secure and stable, as some are more bad than good! Also, be careful with false patches, as every now and then you will find someone creating a Trojan horse patch and trying to pass it off as the real thing.

Therefore, you must maintain your firewall system regularly. Doing so you will be conducting two types of maintenance: a preventive and a curative one. The preventive maintenance is the one you do to play safe, the one ruled by Murphy's law ("whatever bad can happen to the system WILL happen"). The curative one will be done to resolve a problem, to cure a security hole or a flaw in the system's code and so forth. Usually this is done when the vendor releases a new patch, when you have a system's corruption, due to nature disaster, etc., or if the firewall is compromised, as a result of an attack.

The following is a list of good habits, steps and procedures you should follow in order to keep your firewall working properly, which includes both preventive and curative measures:

1. **Back up all firewall components**, not only the bastion host(s), loaded with firewall software, but also the routers.
2. **Make sure when adding new management accounts on a firewall**, as it's very important to maintain your firewall system secure at all times. New accounts must be added correctly, as well as old accounts removed, and make sure to change passwords after deleting an user account. My recommendation is that you should limit the number of user accounts on the firewall, only allowing administrators to access it.
3. **Watch the log reports of traffic passing through the firewall**. Data always expands to fill all



available space, even on machines that have almost no users. Unfortunately, there is no automatic way to find junk on the disk. Auditing programs, like Tripwire, will alert on new files that appear in supposedly static areas. The main disk space problem will be firewall logs. These can and should be rotated automatically with old logs being stored for a minimum of one year.

**4. Monitors your system.** By creating a habit of monitoring your system you will be able to determine several things:

- Has the firewall been under any form of attack?
- If so, what kinds of attacks are being tried against the firewall?
- Is the firewall holding up to this attacks, in working correctly?
- Is the firewall able to provide the service users need?
- Monitor attempts to use the services you disable.
- Configure your system so that any activities related to security is recorded on a log report.
- If your firewall doesn't provide an auditing software, install one, such as Tripwire or L5, run it regularly to spot unexpected changes to your system.
- Log your most critical events to hardcopy if at all possible, and check your logs frequently! Your logs are critical. Most of the time you won't find nothing fun in there, but maybe one of these days you may find an evidence that something is wrong, and you will be thankful to yourself for having coped with this ordeal of checking boring logs.

**1. Be on alert for abnormal conditions of your firewall.** Develop a security checklist, watching for:

- All packets that were dropped
- Denied connections, as well as rejected attempts.
- Data such as time, protocol, and user name of all successful connection to or through the firewall.
- All error messages from your routers, firewalls and any proxying programs.
- Exceptions based on your normal firewall activity. Figure 12.01 outlines a basic access policy.

## ***Preventing Security Breaches on Your Firewall***

Many application level firewalls are already built on the premise that preventing network security problems from occurring in the first place is the best way to resolve them. That should be also your philosophy when implementing and maintaining one.

Several firewall vendors sees prevention as such an important aspect that many are including a security scanning system. Technologic's Interceptor (<http://www.tlogic.com>), for instance, is one of them. Rather than investing in expensive outside security audits, or performing time-consuming internal verifications, with Interceptor, you can confirm that the firewall is doing its job through an Internet Scanner from Internet Security Systems, Inc., which barrages the firewall with simulated break-in attempts. The Internet Scanner is a fast, effective way to verify that Interceptor is configured correctly, and that no security weaknesses have been overlooked.

As outlined above, another preventive maintenance you should periodically perform is to create security-checking reports. This reports can further assist you in identifying potential security problems

with your system. Most of the firewall listed on chapter 14, "Types of Firewalls," produces detailed audit logs of all network traffic activity, as well as other easy-to-read management reports on network access and usage. By regularly reviewing these reports, you will become familiar with network usage patterns, and will be able to recognize aberrations that even hint at trouble.

## Identifying Security Holes

An important first step you should take during the implementation of a firewall is to establish a security policy that defines acceptable use. As discussed on chapter 10, "Putting It Together: Firewall Design and Implementation," very frequently, enforcing your security policy and measuring its effectiveness is usually next to impossible. When new machines or applications are configured, the security related issues are often overlooked. Therefore, the gap between central policy and decentralized practice can be immense. This is what I refer to as the security holes, which can also be generated by bugs in the OS or application.

Remember: If you can measure your security risks, you can control them. Effective control of security risks can only be implemented by assessing the network's security profile. The process of auditing security, correcting vulnerabilities and continuously monitoring activities can close at least most of the security holes not related to the OS or depending on patches.

## Recycling Your Firewall

Just like every other component or device in your network, firewalls also need to be updated so that they can continue to perform and respond to new threads.

Not that you should be pessimist, but if you consider your firewall solution out of date the day you install it, you will be more able to cope with the constant need to update and cover new services under your firewall, some times, especially if you have a packet filtering firewall, you may even need to recycle it.

Of course, you need access to Internet mail and newsgroups, vendors, and other users to be a part of the dialog about changes in network security practices. You also need executive support to be able to avoid having to jump on the bandwagon for every new system that comes along. Just as with application upgrades, it is not necessary to add a new service to your network the day it is issued from the vendors. It is safer to wait and watch a bit while the market "shakes out" the bugs and new security strategies develop. But without a doubt, your firewall is not forever, and eventually you will need to recycle it, update it to say the least.

In the case of system failure other than due to intentional human actions, the main goal in bring the firewall back should be a threefold: 1) to ensure the persistence, and 2) to secure all information; and 3) to facilitate restart. In your organization, accomplishing these goals is achieved by maintaining backup files of all information on a replicate server, located inside the firewall.

[Orders](#)[Orders](#)[Back](#)[Forward](#)

|       |          |
|-------|----------|
| Comme | Comments |
|-------|----------|

---

[COMPUTING MCGRAW-HILL](#) | [Beta Books](#) | [Contact Us](#) | [Order Information](#) | [Online Catalog](#)

---

HTML conversions by [Mega Space](#).

This page updated on December 05, 1997 by [Webmaster](#).

[Computing McGraw-Hill](#) is an imprint of the [McGraw-Hill Professional Book Group](#).

Copyright ©1997 [The McGraw-Hill Companies, Inc.](#) All Rights Reserved.

Any use is subject to the rules stated in the [Terms of Use](#).

|        |        |
|--------|--------|
| Orders | Orders |
|--------|--------|

|      |         |
|------|---------|
| Back | Forward |
|------|---------|

|          |          |
|----------|----------|
| Comments | Comments |
|----------|----------|

© 1997 The McGraw-Hill Companies, Inc. All rights reserved.  
Any use of this Beta Book is subject to the rules stated in the [Terms of Use](#).

---

# Chapter 13

## *Firewall Toolkits And Case Studies*

### The TIS Internet Firewall Toolkit

The TIS Internet Firewall Toolkit is a set of programs and configuration practices designed to facilitate the building of network firewalls. Components of the toolkit, while designed to work together, can be used in isolation or can be combined with other firewall components. The toolkit software is designed to run on UNIX systems using TCP/IP with a Berkeley-style "socket" interface.

I recommend you to access TIS URL at <http://www.tis.com/docs/products/fwtk/fwtkoverview.html> and download the complete document from where this section was extracted. Throughout that documentation, a distinction is made between "configuration practices" and software. A configuration practice is a specific way of configuring existing system software, while a software component of the toolkit is a separate program which may replace or enhance existing system software.

Therefore, when the documentation refers to the configuration practice applicable to configuring some system daemon in a secure manner, it is assumed that the base operating system in question has existing support for that software, and that it is capable of being configured. The exact details of how to configure various system utilities differ from vendor implementation to vendor implementation and are outside of the scope of this document. In general, most UNIX systems with BSD-style networking will support all the functionality and services referred to herein.

Installing the toolkit assumes that you have practical experience with UNIX systems administration and TCP/IP networking. At a minimum, a firewall administrator should be familiar with installing software and maintaining a running UNIX system. Since components of the toolkit are released in source code form, familiarity with building packages using make is required.

The toolkit does not try to provide a "turnkey" network firewall, since every installation's requirements, network topology, available hardware, and administrative practices are different. Depending on how the toolkit is configured, different levels of security can be achieved. The most rigorous security configurations are not for everyone, while for others anything less will not suffice. It is the responsibility of the firewall installer to understand the security policy of the network that is to be protected[1], to understand what constitutes acceptable and unacceptable risks, and to rationalize them with the

requirements of the end users. Performing this analysis is the hardest task in implementing any security system. The toolkit, unfortunately, cannot do it for you; it can only provide the components from which to assemble a solution.

The toolkit consists of three basic components, which are all discussed on that paper:

- Design Philosophy
- Configuration Practices / Verification Strategies
- Software Tools

If you decide on using the toolkit you may use any or none of these components, as you see fit.

Good luck!

## Case Studies: Implementing Firewalls

What kind of firewall is right for your organization? Truthfully, there is not a correct and definite answer. A security policy developed by a Fortune 500 company certainly will not be suitable for a small business owner that yet needs this level of protection on his day-to-day business.

The following are brief scenarios, fictitious, just to illustrate what this book has tried to accomplish as far as enhancing the Internet security of your users and protecting your company's assets from the wild Internet.

### Firewalling a Big Organization: Application-Level Firewall and Package Filtering, a Hybrid System

Other Peoples Money, Inc. (OPM), might find an application-level firewall (see chapter 14, "Types of Firewall Products") and packet filters adequate for its needs, based on its outgoing access capabilities and high, diversified income traffic. OPM may even consider creating a DMZ zone where they Web server, RASing connectivity and other not so public services would be secured, but not to the point of compromising the protected LAN.

It would be advisable to implement CERT's recommendation of an additional router to filter and block all packets whose addresses are originated from inside the protected network. This two-router solution is not complicated to deploy, and is very cost-effective when you consider that OPM would be exposed to spoofing by allowing all 600 employees throughout the country to have access to its Web server and internal network.

When implementing two routers, OPM should purchase them from different companies (that is, choose two different brands). It might sound like nonsense, but if a hacker is able to break into one router due to a bug or a back door on the router's code, the second router will not have the same codes. Even though the firewall will no longer be transparent, which will require users to log on to it, the site will be protected, monitored, and safe. The two routers create a package-filtering firewall while the bastion host functions as an application-gateway firewall, where the software will be installed.

# Firewalling a Small Organization: Packet Filtering or Application-Level Firewall, a Proxy Implementation

The same model used at OPM would suit Dry Water Co., although the policy would be much simpler than OPM's.

A proxy server, or one of the firewalls listed on chapter 14 would be enough. GNET's firewall on a diskette would probably be the most adequate.

This configuration would assume that all the department and internal organizations trust each other and the internal users. But even if they didn't every users could have their own GNET at their computer.

## Firewalling in a Subnet Architecture

If you decide to protect subnets within your organization, the IP-level filtering might be the most appropriate versus other types of filtering. This model enables each type of client and service basically to be supported within the internal network.

No modifications or special client software would be necessary. The access through the IP-level filtering firewall will be totally transparent for the user and the application. The existing router can be utilized for the implementation of the IP-level filtering. There will be no need to buy an expensive UNIX host or firewall product. But again, as you review the products outlined on chapter 14, you may decide to filter those subnet connection using application-level solutions, especially if you don't have the hardware necessary.

|          |          |
|----------|----------|
| Orders   | Orders   |
| Comments | Comments |

|      |         |
|------|---------|
| Back | Forward |
|------|---------|

---

[COMPUTING MCGRAW-HILL](#) | [Beta Books](#) | [Contact Us](#) | [Order Information](#) | [Online Catalog](#)

---



HTML conversions by [Mega Space](#).

This page updated on December 05, 1997 by [Webmaster](#).

[Computing McGraw-Hill](#) is an imprint of the [McGraw-Hill Professional Book Group](#).

Copyright ©1997 [The McGraw-Hill Companies, Inc.](#) All Rights Reserved.  
Any use is subject to the rules stated in the [Terms of Use](#).

Orders

Orders

Back

Forward

Comments

Comments

© 1997 The McGraw-Hill Companies, Inc. All rights reserved.  
Any use of this Beta Book is subject to the rules stated in the [Terms of Use](#).

## Chapter 14

# *Types of Firewalls and Products on the Market*

This section provides you with a technical overview of the main firewall products available on the market as of Summer of 1997. I made sure to include a vast and extensive selection of all the major players and architecture so you can have a chance to evaluate each one of them before deciding which firewall best suite your needs.

This selection includes many different firewall architectures, from application proxy and circuit relay ones, such as Raptor's EagleNT, Ukiah's NetRoad and Secure Computing's Borderware firewall, to stateful inspection and packet filter ones, such as WatchGuard Technologies's WatchGuard, Sun's SunScreen, Check Point's Firewall-1 and Cycon's Labyrinth ones.

Evidently, I'm not in the position of recommending any of these products as the needs and features of a firewall product will change depending of your environment. Although I may have my preferences, it probably would be a biased one, which would be directly related to the environment I work with. Thus, all the information you find in this section was totally provided by the vendor of each firewall outlined here. Some provided more information then others, as others provided more graphics and figures. By no means you should opt for any of these firewalls based on the amount of pages or details here provided. Most of the vendors listed here also provided demo and/or evaluation copies of their products in the CD that accompanies this book.

In order to make an informative decision when selecting a firewall that best suites your needs, I strongly encourage you to carefully read this chapter, and summarize on a table all the features you are looking for, or need, in a firewall for your organization. Then, I suggest you to check the CD and install the firewall(s) you selected and run a complete "dry-run" on them before you can really make a decision. Also, don't forget to contact the vendor directly, as these products are always being upgraded and new features incorporated to them, which could make a difference in your decision. Contact information and a brief background about the vendor is provided at the beginning of every section of the product covered.

## Check Points' Firewall-1 Firewall - Stateful Inspection Technology

Check Point FireWall-1, developed by Check Point Software Technologies (<http://www.checkpoint.com>), is based upon Stateful Inspection architecture, the new generation of firewall technology invented by CheckPoint. Stateful Inspection Technology delivers full firewall capabilities, assuring the highest level of network security. FireWall-1's powerful Inspection Module analyzes all packet communication layers, and extracts the relevant communication and application state information. The Inspection Module understands and can learn any protocol and application. Figure 14.1 shows a screenshot of Check Points site.

### Note:

For more information, contact Check Point Software Technologies, Redwood City, CA, (415) 562-0400 or at their Web site at URL <http://www.checkpoint.com>



# ***FireWall-1 Inspection Module***

The FireWall-1 Inspection Module resides in the operating system kernel, below the Network layer, at the lowest software level. By inspecting communications at this level, FireWall-1 can intercept and analyze all packets before they reach the operating systems. No packet is processed by any of the higher protocol layers unless FireWall-1 verifies that it complies with the enterprise security policy.

## **Full State Awareness**

The Inspection Module has access to the "raw message," and can examine data from all packet layers. In addition, FireWall-1 analyzes state information from previous communications and other applications. The Inspection Module examines IP addresses, port numbers, and any other information required in order to determine whether packets comply with the enterprise security policy. It also stores and updates state and context information in dynamic connections tables. These tables are continually updated, providing cumulative data against which FireWall-1 checks subsequent communications. FireWall-1 follows the security principle of "All communications are denied unless expressly permitted." By default, FireWall-1 drops traffic that is not explicitly allowed by the security policy and generates real-time security alerts, providing the system manager with complete network status.

## **Securing "Stateless" Protocols**

The FireWall-1 Inspection Module understands the internal structures of the IP protocol family and applications built on top of them. For stateless protocols such as UDP and RPC, the Inspection Module extracts data from a packet's application content and stores it in the state connections tables, providing context in cases where the application does not provide it. In addition, it can dynamically allow or disallow connections as necessary. These capabilities provide the highest level of security for complex protocols.

## **The INSPECT Language**

Using Check Point's INSPECT language, FireWall-1 incorporates security rules, application knowledge, context information, and communication data into a powerful security system. INSPECT is an object-oriented, high-level script language that provides the Inspection Module with the enterprise security rules.

In most cases, the security policy is defined using FireWall-1's graphical interface. From the security policy, FireWall-1 generates an Inspection Script, written in INSPECT. Inspection Code is compiled from the script and loaded on to the FireWalled enforcement points, where the Inspection Module resides. Inspection Scripts are ASCII files, and can be edited to facilitate debugging or meet specialized security requirements.

INSPECT provides system extensibility, allowing enterprises to incorporate new applications, services, and protocols simply by modifying one of FireWall-1's built-in script templates using the graphical user interface. Figure 14.2 shows a diagram of the Stateful Inspection technology.

## ***Stateful Inspection: Under the hood***

As discussed throughout this book, in order for you to have a robust security at your company you should have a firewall. But this firewall must be able to track and control the flow of communication passing through it. To reach control decisions for TCP/IP based services, such as whether to accept, reject, authenticate, encrypt and/or log communication attempts, a firewall must obtain, store, retrieve and manipulate information derived from all communication layers and from other applications.

It is not sufficient to examine packets in isolation. State information, which is derived from past communications and other applications, is an essential factor in making the control decision for new communication attempts. Depending upon the communication attempt, both the communication state, derived from past communications, and the application state, derived from other applications, may be critical in the control decision.

Thus, to ensure the highest level of security, a firewall must be capable of accessing, analyzing and utilizing the following:

- **Communication Information** - information from all seven layers in the packet
- **Communication-derived State** - the state derived from previous communications. For example, the outgoing PORT command of an FTP session could be saved so that an incoming FTP data connection can be verified against it.
- **Application-derived State** - the state information derived from other applications. For example, a previously authenticated user would be allowed access through the firewall for authorized services only.
- **Information Manipulation** - the evaluation of flexible expressions based on all the above factors.

Check Point's Stateful Inspection is able to meet all the security requirements defined above. Traditional firewall technologies, such as packet filters and application-layer gateways, each fall short in some areas, as shown on Table I.

**Table I: Comparison of capabilities for three main firewall architectures**

| Firewall Capability         | Packet Filters | Application-layer Gateways | Stateful Inspection |
|-----------------------------|----------------|----------------------------|---------------------|
| Communication Information   | Partial        | Partial                    | Yes                 |
| Communication-derived State | No             | Partial                    | Yes                 |
| Application-derived State   | No             | Yes                        | Yes                 |
| Information Manipulation    | Partial        | Yes                        | Yes                 |

If you take packet filters, for example, historically they are implemented on routers, are filters on user defined content, such as IP addresses. As discussed on chapter 7, "What is an Internet/Intranet Firewall After All?," packet filters examine a packet at the network layer and are application independent, which allows them to deliver good performance and scalability. However, they are the least secure type of firewall, especially when filtering services such as FTP, which was vastly discussed on chapter 8, "How Vulnerable Are Internet Services." The reason is that they are not application aware-that is, they cannot understand the context of a given communication, making them easier for hackers to break. Figure 14.3 illustrates it.

If we look into FTP filtering, packet filters have two choices with regard to the outbound FTP connections. They can either leave the entire upper range (greater than 1023) of ports open which allows the file transfer session to take place over the dynamically allocated port, but exposes the internal network, or they can shut down the entire upper range of ports to secure the internal network which blocks other services, as shown on figure 14.4. This trade-off between application support and security is not acceptable to users today.

As with application gateways, as shown on figure 14.5, the security is improved by examining all application layers, bringing context information into the decision process. However, they do this by breaking the client/server model. Every client/server communication requires two connections: one from the client to the firewall and one from the firewall to the server. In addition, each proxy requires a different application process, or daemon, making scalability and support for new applications a problem.

For instance, in using an FTP proxy, the application gateway duplicates the number of sessions, acting as a proxied broker between the client and the server (see figure 14.6). Although this approach overcomes the limitation of IP filtering by bringing application-layer awareness to the decision process, it does so with an unacceptable performance penalty. In addition, each service needs its own proxy, so the number of available services and their scalability is limited. Further, this approach exposes the operating system to external threats.

The Stateful Inspection introduced by Check Point overcomes the limitations of the previous two approaches by providing full application-layer awareness without breaking the client/server model. With Stateful Inspection, the packet is intercepted at the network layer, but then the INSPECT Engine takes over, as shown on figure 14.7. It extracts state-related information required for the security decision from all application layers and maintains this information in dynamic state tables for evaluating subsequent connection attempts. This provides a solution which is highly secure and offers maximum

performance, scalability, and extensibility.

The Stateful Inspection tracks the FTP session, as shown on figure 14.8, examining FTP application-layer data. When the client requests that the server generate the back-connection (an FTP PORT command), FireWall-1 extracts the port number from the request. Both client and server IP addresses and both port numbers are recorded in an FTP-data pending request list. When the FTP data connection is attempted, FireWall-1 examines the list and verifies that the attempt is in response to a valid request. The list of connections is maintained dynamically, so that only the required FTP ports are opened. As soon as the session is closed the ports are locked, ensuring maximum security.

## Extensible Stateful Inspection

Check Point FireWall-1's Stateful Inspection architecture utilizes a unique, patented INSPECT Engine which enforces the security policy on the gateway on which it resides. The INSPECT Engine looks at all communication layers and extracts only the relevant data, enabling highly efficient operation, support for a large number of protocols and applications, and easy extensibility to new applications and services.

The INSPECT Engine is programmable using Check Point's powerful INSPECT Language. This provides important system extensibility, allowing Check Point, as well as its technology partners and end-users, to incorporate new applications, services, and protocols, without requiring new software to be loaded. For most new applications, including most custom applications developed by end users, the communication-related behavior of the new application can be incorporated simply by modifying one of FireWall-1's built-in script templates via the graphical user interface. Even the most complex applications can be added quickly and easily via the INSPECT Language.

### Tip:

Check Point provides an open application programming interface (API) for third-party developers and regularly posts INSPECT Scripts to support new applications on the Check Point Web site at <http://www.checkpoint.com>.

## The INSPECT Engine

When installed on a gateway, the FireWall-1 INSPECT Engine controls traffic passing between networks. The INSPECT Engine is dynamically loaded into the operating system kernel, between the Data Link and the Network layers (layers 2 and 3). Since the data link is the actual network interface card (NIC) and the network link is the first layer of the protocol stack (for example, IP), FireWall-1 is positioned at the lowest software layer. By inspecting at this layer, FireWall-1 ensures that the INSPECT Engine intercepts and inspects all inbound and outbound packets on all interfaces. No packet is processed by any of the higher protocol stack layers, no matter what protocol or application the packet uses, unless the INSPECT Engine first verifies that the packet complies with the security policy.

As discussed earlier, because the INSPECT Engine has access to the "raw message", it can inspect all the information in the message, including information relating to all the higher communication layers, as well as the message data itself (the communication- and application-derived state and context). The INSPECT Engine examines IP addresses, port numbers, and any other information required in order to determine whether packets should be accepted, in accordance with the defined security policy.

The INSPECT Engine understands the internal structures of the IP protocol family and applications built on top of them. For stateless protocols such as UDP and RPC, the INSPECT Engine creates and stores context data, maintaining a virtual connection on top of the UDP communication. The INSPECT Engine is able to extract data from the packet's application content and store it to provide context in those cases where the application does not provide it. Moreover, the INSPECT Engine is able to dynamically allow and disallow connections as necessary. These dynamic capabilities are designed to provide the highest level of security for complex protocols, but the user may disable them if they are not required.

The INSPECT Engine's ability to look inside a packet enables it to allow certain commands within an application while disallowing others. For example, the INSPECT Engine can allow an ICMP ping while disallowing redirects, or allow SNMP gets while disallowing sets, and so on. The INSPECT Engine can store and retrieve values in tables (providing dynamic

context) and perform logical or arithmetic operations on data in any part of the packet. In addition to the operations compiled from the security policy, the user can write his or her own expressions.

Unlike other security solutions, FireWall-1's Stateful Inspection architecture intercepts, analyzes, and takes action on all communications before they enter the operating system of the gateway machine, ensuring the full security and integrity of the network. Cumulative data from the communication and application states, network configuration and security rules, are used to generate an appropriate action, either accepting, rejecting, authenticating, or encrypting the communication. Any traffic not explicitly allowed by the security rules is dropped by default and real-time security alerts and logs are generated, providing the system manager with complete network status.

The Stateful Inspection implementation supports hundreds of pre-defined applications, services, and protocols, more than any other firewall vendor. Support is provided for all major Internet services, including secure Web browsers, the traditional set of Internet applications (e.g. mail, FTP, Telnet, etc.), the entire TCP family, and connectionless protocols such as RPC and UDP-based applications. In addition, only FireWall-1's Stateful Inspection offers support for critical business applications such as Oracle SQL\*Net database access and emerging multimedia applications such as RealAudio, VDOLive, and Internet Phone.

## Securing Connectionless Protocols such as UDP

UDP (User Datagram Protocol)-based applications (DNS, WAIS, Archie, etc.) are difficult to filter with simplistic packet-filtering techniques because in UDP, there is no distinction between a request and a response. In the past, the choice has been to either eliminate UDP sessions entirely or to open a large portion of the UDP range to bi-directional communication, and thus to expose the internal network.

Stateful Inspection implementation secures UDP-based applications by maintaining a virtual connection on top of UDP communications. The FireWall-1's INSPECT Engine maintains state information for each session through the gateway. Each UDP request packet permitted to cross the firewall is recorded, and UDP packets traveling in the opposite direction are verified against the list of pending sessions to ensure that each UDP packet is in an authorized context. A packet that is a genuine response to a request is delivered and all others are dropped. If a response does not arrive within the specified time period, the connection times out. In this way, all attacks are blocked, while UDP applications can be utilized securely.

## Securing Dynamically Allocated Port Connections

Simple tracking of port numbers fails for RPC (Remote Procedure Call) because RPC-based services (NFS, NIS) do not use pre-defined port numbers. Port allocation is dynamic and often changes over time. This is another feature of the INSPECT Engine of Firewall-1, which dynamically and transparently tracks RPC port numbers using the port mappers in the system. The INSPECT Engine tracks initial portmapper requests and maintains a cache that maps RPC program numbers to their associated port numbers and servers.

Whenever the INSPECT Engine examines a rule in which an RPC-based service is involved, it consults the cache, comparing the port numbers in the packet and cache and verifying that the program number bound to the port is the one specified in the rule. If the port number in the packet is not in the cache (this can occur when an application relies on prior knowledge of port numbers and initiates communication without first issuing a portmapper request) the INSPECT Engine issues its own request to portmapper and verifies the program number found to the port, as shown on figure 14.9.

## Firewall-1 Performance

The following are the major performance strength of Firewall on through its INSPECT Engine:

- Runs inside the operating-system kernel, which imposes negligible overhead in processing. Also, no context switching is required, and low-latency operation is achieved.
- Uses of advanced memory management techniques, such as caching and hash tables, which are used to unify multiple object instances and to efficiently access data.
- Its generic and simple inspection mechanisms are combined with a packet inspection optimizer, which ensure optimal utilization of modern CPU and OS designs.

According to independent test results (<http://www.checkpoint.com/products/fproduct.html>) and an article on Data Communication magazine of March 97, the network performance degradation when using Firewall-1 is too small to measure when operating at full LAN speed (10 Mbps) on the lowest-end SPARCstation. FireWall-1 supports high-speed networking such as 100 Mbps Ethernet and OC-3 ATM with the same high level of performance.

As far as certified benchmark, KeyLabs Inc. (<http://www.keylabs.com>) conducted extensive testing of the Solaris and Windows NT versions of FireWall-1 to document firewall performance under various configurations. The test methodology was carefully designed to simulate actual network conditions and test automation applications were employed to ensure accurate results.

Several FireWall-1 configurations were tested to determine whether performance is impacted by encryption, address translation, logging and rule base size. In addition, FireWall-1 was stressed to determine the maximum number of concurrent connections that can be supported. The Fastpath option was enabled on FireWall-1 for several configurations to maximize performance. Fastpath is a widely used FireWall-1 feature that optimizes performance without compromising security.

FireWall-1 was configured with two network interfaces: internal and external. Each interface utilized two Fast Ethernet connections to maximize throughput and ensure that FireWall-1 was thoroughly stressed. Multiple clients on the internal network made HTTP and FTP requests to multiple servers on the external side of FireWall-1. Clients generated approximately 5 Mbps of traffic each and were added incrementally to increase the traffic level through FireWall-1.

During this test, 75% of the connections were of HTTP (75 kbytes) 75%, and the remain 25% were FTP (1 Mbyte) connections.

To determine the maximum number of concurrent connections that FireWall-1 can support, multiple clients made HTTP requests to servers on the external FireWall-1 interface. Each client was capable of establishing and maintaining 500 total connections, as shown on figure 14.10.

The results? When running on Solaris, as shown on figure 14.11, FireWall-1 supports approximately 85 Mbps with Fastpath enabled (top line) and 53 Mbps with Fastpath disabled (second line from top). This is sufficient to support both T3 (45 Mbps) and effective Fast Ethernet data rates.

For Windows NT, as shown on figure 14.12, 25 Mbps can be maintained with Fastpath enabled, and approximately 20 Mbps is supported without Fastpath. This is seen in the bottom two lines of the graph. The test results show that both T1 (1.544 Mbps) and Ethernet data rates are supported by the Windows NT version of FireWall-1. With this level of performance across multiple platforms, FireWall-1 is well-suited for high-speed Internet and Intranet environments.

For more information, check Keylabs Inc. Site, as listed above or Check Point's site. There you will find a comprehensive result of Firewall-1 performance in many other environment and situations.

## Systems Requirements

The FireWall-1 system requirements is the following:

- **Platforms supported:** Sun SPARC, HP-PA-RISC 700/800, Intel x86 or Pentium
- **Operating systems:** Windows NT 3.51 and 4.0, SunOS 4.1.3 and 4.1.4, Solaris 2.3, 2.4, and 2.5, HP-UX 9 and 10 and IBM AIX
- **Window systems:** Windows 95, Windows NT, X/Motif and Open Look
- **Disk space:** 20 MB
- **Memory:** 16-32 MB
- **Network interface:** All interfaces supported by the operating system
- **Routers management (optional):** Cisco Systems IOS version 9, 10, 11 Bay Networks version 8, 9
- **Media:** CD-ROM

# CYCON's Labyrinth Firewall - The "Labyrinth-like" System

The CYCON Labyrinth firewall is the world's first "labyrinth-like" system incorporating true bi-directional network address translation with a powerful, intelligent connection tracking (ICT) firewall to create an integrated security and network management device. CYCON Labyrinth firewall is currently in use by several major corporations, Internet Service Providers, and research institutions. Figure 14.13 shows a screenshot of CYCON's site.

**Note:**

For more information, contact CYCON Technologies, Fairfax, VA, (703) 383-0247, or at their Web site at URL <http://www.cycon.com>

CYCON Labyrinth firewall's stateful inspection engines support all IP based services and correctly follows TCP, UDP, ICMP, and TCP SYN/ACK traffic. Support for all major IP services include, but not limited to:

- Telnet
- SMTP
- DNS (both TCP and UDP)
- FTP
- HTTP
- SSL
- NFS
- NNTP
- Archie
- Gopher
- X11
- NTP
- X500
- IMAP and POP3
- LDAP
- ICMP (ping, traceroute)
- RealAudio

CYCON Labyrinth firewall offers full bi-directional network address translation. CYCON Labyrinth firewall can rewrite the source, destination, and port addresses of a packet. Network address translation conceals internal addresses from outside untrusted networks. Additionally, bi-directional address translation enables CYCON Labyrinth firewall to properly redirect packets to any host in any system. Using two CYCON Labyrinth firewalls together allow the proper communication between two private IP networks connected to the Internet by translating both incoming and outgoing traffic.

CYCON Labyrinth firewall can be configured to authenticate users on both inbound and outbound access. Inbound access authentication is used to implement stronger security policies. Outbound access authentication can be used to track and log connections for internal billing or charge backs purposes. Authentication is at the user level, not at the IP address level. This allows the user to move across networks and retain the ability to use resources regardless of their physical IP address, making it appropriate for Dynamic Host Configuration Protocol (DHCP) address assignments.

CYCON Labyrinth firewall supports multi-level logging. In regular mode, connections are logged. In debug logging mode, connections, packets, bytes, and actions taken are logged. Log files are written in standard UNIX syslog ASCII format and are easily manipulated by a firewall administrator for analysis. Syslog logging allows multiple CYCON Labyrinth firewalls to log to a single machine for greater security and ease of analysis.

CYCON Labyrinth firewall utilizes a rewritten BSD UNIX kernel incorporating optimized data structures and algorithms

designed to produce high-speed packet filtering. CYCON Labyrinth firewall implements stateful inspection and packet modifying technology to overcome gaps found in traditional packet filtering methods

## ***An Integrated Stateful Inspection***

The CYCON Labyrinth firewall provides outstanding protection to all aspects of an organization's network: Internet, Intranet, and enterprise-wide connectivity. Its security model utilizes next generation firewall technology, intelligent tracking of connections, and packet modifying engines to offer transparent use of current and emerging Internet technologies. Client applications and protocol stacks operate without modifications.

Features include user authentication, high-speed static and dynamic filters, Web-based management GUI, support for up to six network interfaces, real-time monitoring and reporting, multiple logging levels, and custom alarm notifications. Also, introduced in January of 1997, this firewall includes IPSec-compliant encryption for Virtual Private Networking, native low port NFS support, and secure support for VDO, Vosaic, VXTreme, Internet Phone, and Microsoft's NetShow.

Its integrated stateful inspection and bi-directional network address translation engines allows this firewall to perform unusual tasks, as discussed below. Also, command line syntax is included in the descriptions, as appropriate, to explain how the CYCON Labyrinth firewall's filter rules can be applied in each scenario.

Here are some examples of these tasks:

- Intelligent Connection Tracking
- Redirecting Traffic
- Network Address Translation
- Load Balancing of Connections
- Proxying - source address rewriting
- Spoofing - destination address rewriting
- IPSec - encryption

### **Intelligent Connection Tracking**

The CYCON Labyrinth firewall transcends ordinary packet filtering devices by utilizing an advanced connection tracking feature called Intelligent Connection Tracking (ICT). The ICT module is designed to recognize the internal portions of IP packets, enabling the CYCON Labyrinth firewall to "remember" authorized connections for replies. This algorithm heightens security, as it alleviates opening large holes in the filter rules required by other firewalls. These large holes are common areas of security breaches.

The ICT module examines each packet as it is processed by the CYCON Labyrinth firewall, and stores vital information about the packet. The module compares the packet information to the saved state of previously transmitted packets, and permits only those packets that are successfully tested.

The CYCON Labyrinth firewall uses a combination of static and dynamic filter rules to achieve ICT. As traffic crosses an interface, it encounters one of two rule sets: dynamic or static. If a static rule considered Stateful is encountered, the CYCON Labyrinth firewall creates a dynamic rule in the opposite direction, allowing the manipulations performed on the outbound packet to be reversed when the destination system replies. A stateful static rule would be:

```
ipcycon de0 in spoof ip 0.0.0.0:0.0.0.0 1.1.1.1 spoofaddr 2.2.2.2
```

Traffic enters the interface "de0" from any source address destined for the host 1.1.1.1. These packets match static rules and trigger the following dynamic rule on the outbound portion of the interface:

```
ipcycon de0 out proxy ip 2.2.2.2 3.3.3.3 spoofaddr 1.1.1.1
```

When traffic returns from host 2.2.2.2 destined for host 3.3.3.3, it will match the above dynamic rule and adjust the source

address back to 1.1.1.1 so traffic will route normally.

This example illustrates how the CYCON Labyrinth firewall "remembers" the packet state via the dynamic static rule couplet and can successfully route the traffic through the ICT algorithm.

## Redirecting Traffic

The address translation features of the CYCON Labyrinth firewall is used to redirect traffic to any host in any network. This feature has a number of real-world uses; examples include: transparent redirection to fault-tolerant systems; diverting scanning programs back to the attacker; and diverting an attacker to a dummy machine specifically designed to trap and log the attacker.

Address translation is accomplished by altering the destination address of the packet to a new location, and altering the source address of the packet to reflect the address of the CYCON Labyrinth firewall. When the reply packets are returned from the receiving host, the address translation process is reversed, and the packets are rewritten and sent to the original sender as though they came from the originally intended destination.

### *Transparent Redirection to Fault-Tolerant Systems*

If a Web server inside your network fails, the CYCON Labyrinth system is used to redirect all incoming Web traffic to an external backup system. This is accomplished by changing both the source and destination addresses in the packets destined for your internal Web server. Assuming your failed Web server's IP address is 1.1.1.1 and the backup external Web server's IP address is 2.2.2.2, the exact rules to accomplish this are:

```
ipcycon de0 in spoof tcp 0.0.0.0:0.0.0.0 1.1.1.1:255.255.255.255 dst-eq 80 spoofaddr
2.2.2.2:255.255.255.255
```

```
ipcycon de0 out proxy tcp 0.0.0.0:0.0.0.0 2.2.2.2:255.255.255.255 dst-eq 80
```

The second command alters the source address, ensuring that replies from the external Web server are sent through CYCON Labyrinth system and back to the client.

### *Diverting Scanning Programs*

A special variation of the spoof rule redirects unwanted traffic back to the sender. This particular form of the spoof rule is called Rubber and Glue, and is designed specifically to confuse hackers by reversing the connection back onto the hackers system(s). For example, to redirect all unwanted traffic entering your network back to the sender, use the following spoof rule (assuming your network is 1.1.1.0):

```
ipcycon de0 in spoof ip 0.0.0.0:0.0.0.0 1.1.1.0:255.255.255.0 spoofaddr
0.0.0.0:255.255.255.255
```

The source address of these unwanted packets will need to be altered to complete the ruse, as follows:

```
ipcycon de0 out proxy ip 0.0.0.0:0.0.0.0 0.0.0.0:0.0.0.0
```

This spoof rule uses the 0.0.0.0:255.255.255.255 as the spoof address. This special form of the spoof address is used to represent the source address of the original packets. The result is that the packet's destination address is changed to the source address, and the source address is changed to the firewall's address. All of these address changes are reversed when a reply packet is received.

A more complex version of this rule may be used to redirect traffic to the hacker's network instead of redirecting traffic to the source address used in the attack. This is accomplished by using a different mask on the spoof address. For example, to



redirect unwanted traffic to the hacker's network, use the following spoof rule:

```
ipcycon de0 in spoof ip 0.0.0.0:0.0.0.0 1.1.1.0:255.255.255.255.0 spoofaddr
0.0.0.0:255.255.255.0
```

This rule will cause the destination address to be replaced with the first three octets of the hacker's address, followed by the last octet of the original destination. If the hacker is coming from the address 3.3.3.3 and sending packets to 1.1.1.15, then the new destination address would be 3.3.3.15. If the hacker is sending packets to 1.1.1.32, then new destination address would be 3.3.3.32.

## Network Address Translation

The CYCON Labyrinth firewall transcends traditional Network Address Translation (NAT) by utilizing full bi-directional network address translation. Ordinary NAT is used to alter either the source or destination address in packet headers. Bi-directional NAT rewrites source, destination, and port identifiers of packets on both inbound and outbound interface traffic. This is extremely useful to route traffic over public networks using private IP addresses or to load balance one URL among multiple Web or FTP servers.

The CYCON Labyrinth firewall performs bi-directional NAT by using special rules to instruct the translation modules to substitute any address "B" for the originating address "A" of a packet. The syntax of the command would appear as follows:

```
ipcycon de0 out proxy ip 1.1.1.1 165.80.1.1 spoofaddr 192.80.4.3
```

|             |  |  |  |  |  |  |  |  |  |  |
|-------------|--|--|--|--|--|--|--|--|--|--|
|             |  |  |  |  |  |  |  |  |  |  |
| command     |  |  |  |  |  |  |  |  |  |  |
| interface   |  |  |  |  |  |  |  |  |  |  |
| direction   |  |  |  |  |  |  |  |  |  |  |
| action      |  |  |  |  |  |  |  |  |  |  |
| service     |  |  |  |  |  |  |  |  |  |  |
| source      |  |  |  |  |  |  |  |  |  |  |
| destination |  |  |  |  |  |  |  |  |  |  |
| tag         |  |  |  |  |  |  |  |  |  |  |
| NAT address |  |  |  |  |  |  |  |  |  |  |

This command alters IP packets leaving the interface "de0" from source 1.1.1.1 bound for destination 165.80.1.1 so that the

source address is rewritten to 192.80.4.3. The CYCON Labyrinth firewall has mechanisms in place for proper translations of any reply packets.

The packet leaving the de0 interface is detected by the CYCON Labyrinth firewall as its internal rules are being processed, and is marked for action because the packet originated from host "1.1.1.1" and is destined for the host "165.80.1.1". As 1.1.1.1 is not routable on the public network, the source address must be changed or the sender will get the error No Route to Host.

If a rule matching the source and destination addresses is encountered, the Proxy action occurs, and the spoofaddr address 192.80.4.3 is substituted for 1.1.1.1 as the source address. The packet is modified and routed through the interface.

This is all that is necessary to route the packet out of the network, but any replies to the packets will have 192.80.4.3 as the destination address. Replies to 192.80.4.3 will not be routed back properly into the internal network, so the CYCON Labyrinth firewall rewrites the incoming destination address. The CYCON Labyrinth firewall remembers the original source address and established port of the packet and rewrites packets of expected reply traffic (known as Intelligent Connection Tracking).

When the original packet was processed and the 1.1.1.1 address rewritten, the CYCON Labyrinth firewall created a dynamic rule and applied it to the inbound portion of the de0 interface, noting the original destination address and destination port of the packet. When the firewall encounters traffic from 165.80.1.1 destined for 192.80.4.3 on port 3456 (in this example, a negotiated TCP port), the CYCON Labyrinth firewall knows to replace the 192.80.4.3 destination address back to 1.1.1.1 and route the packet to the internal network. This dynamic rule remains until the transaction is terminated and removed from memory.

The following is a time-lapse view of how and when the packets are rewritten:

- Packet going out to destination

| source | destination |
|--------|-------------|
|--------|-------------|

|         |            |
|---------|------------|
| 1.1.1.1 | 165.80.1.1 |
|---------|------------|

- Source address being rewritten by the CYCON Labyrinth firewall

| source | destination |
|--------|-------------|
|--------|-------------|

|            |            |
|------------|------------|
| 192.80.4.3 | 165.80.1.1 |
|------------|------------|

- Reply packet coming back from outside host

| source | destination |
|--------|-------------|
|--------|-------------|

|            |            |
|------------|------------|
| 165.80.1.1 | 192.80.4.3 |
|------------|------------|

- Destination address being rewritten by the CYCON Labyrinth firewall

| source | destination |
|--------|-------------|
|--------|-------------|

|            |         |
|------------|---------|
| 165.80.1.1 | 1.1.1.1 |
|------------|---------|

This concept of altering source and destination addresses can be applied to either direction (inbound and outbound) and on any individual interface. This provides extreme flexibility for generating rules. Other examples of the applicability include load-balancing one address among multiple servers, directing any inbound web requests to one web server on the DMZ, and

sending all SATAN packets back to the originator (causing attackers to attack themselves).

## Load Balancing of Connections

The CYCON Labyrinth firewall uses SPOOF and PROXY rules to load balance incoming connections between multiple hosts and/or networks. Load balancing is a process in which packets are redirected to alternating hosts or networks per concurrent connection. This capability allows organizations to use multiple small hosts to serve requests, rather than investing in high-powered systems.

Using standard IP addresses and netmasks, you can construct a single rule that can disperse traffic to four different hosts and networks.

These special rules use a standard rolodex calculation. Each time a connection is established, the firewall directs the connection to the next available address. When the list of addresses has been exhausted, the CYCON Labyrinth firewall returns to the beginning of the list to establish the connection.

### *Multi-Host Load Balancing*

Advertised Address:1.1.1.5

Web Server 1: 1.1.1.1

Web Server 2: 1.1.1.2

Web Server 3: 1.1.1.3

Web Server 4: 1.1.1.4

```
ipcycon de0 in spoof ip 0.0.0.0:0.0.0.0 1.1.1.5:255.255.255.0
```

```
spoofaddr 1.1.1.1 1.1.1.2 1.1.1.3 1.1.1.4
```

The CYCON Labyrinth firewall is intelligent. Utilizing intelligent connection tracking modules, the firewall creates dynamic rules for each connection and thus "remembers" the correct host.

This technology enables an organization to spread connections to one web address across multiple web servers. Without the CYCON Labyrinth firewall, an organization is forced to use either multiple web servers or inefficient round-robin Domain Name Server (DNS) techniques.

## Proxying - Source Address Rewriting

The CYCON Labyrinth firewall offers bi-directional address translation of host and network addresses; that is, the CYCON Labyrinth firewall has the capability to translate addresses in the header portion of IP packets on traffic either entering or leaving a specific interface. This is particularly useful in areas such as host load balancing, using private IP addresses in a public space, hiding internal networks, etc.

CYCON Technologies uses the term Proxy to describe the capability of rewriting the source address of IP packet headers. Proxying IP addresses allows sites to use private or unregistered addresses to connect to the Internet using any publicly routed address, thereby hiding internal IP addresses and eliminating the high cost of reassigning IP addresses when changing providers. Utilizing special rules, the CYCON Labyrinth firewall, upon receiving traffic that matches a proxy rule, rewrites the source address to an individual address, translates network to network, or chose one of four possible network/hosts addresses.

The CYCON Labyrinth firewall utilizes subnetmasks to achieve the host to host, host to network, and network to network address translation. A wild card mask - 0 - can be used in any octet position to cause the firewall to use the existing assumed octet address. If the spoof address is left blank, the address of the interface is assumed.

In the event a site using a private address space wants to access the Internet, the only option in the past was to acquire an IP segment from the provider and visit each host and alter configurations. This is both time consuming and costly. Utilizing the Proxy feature of the CYCON Labyrinth firewall, organizations can get a single Class C address space and proxy all traffic, creating the appearance that it is coming from the provided network. For example:

```
ipcycon de0 out proxy ip 172.16.1.0:255.255.255.0 0.0.0.0:0.0.0.0 spoofaddr
204.5.16.0:255.255.255.0
```

## Spoofing - Destination Address Rewriting

The CYCON Labyrinth firewall offers bi-directional address translation of host and network addresses, that is, the firewall has the capability to translate addresses in the header portion of IP packets on traffic either entering or leaving a specific interface. This is particularly useful in areas such as load balancing, using private IP addresses in a public space, hiding internal networks, etc.

CYCON Technologies uses the term Spoof to describe the capability of rewriting the destination address of IP packet headers. Utilizing special rules, the CYCON Labyrinth firewall, upon receiving traffic that matches a spoof rule, rewrites the destination address to one address, translates network to network, or chooses one of four possible network/hosts addresses.

The CYCON Labyrinth firewall utilizes subnetmasks to achieve the host to host, host to network, and network to network address translation. A wild card mask - 0 - can be used in any octet position to cause the firewall to use the existing destination octet address. The following are examples of the rules:

- **Host to Host** - When the CYCON Labyrinth firewall encounters a packet coming from any host destined for host 1.1.1.1, it changes the 1.1.1.1 address to 2.2.2.2. For example:

```
ipcycon de0 in spoof ip 0.0.0.0:0.0.0.0 1.1.1.1 spoofaddr 2.2.2.2:255.255.255.255
```

- **Host to Network** - When the CYCON Labyrinth firewall encounters a packet coming from any host destined for host 1.1.1.1, it changes the 1.1.1.1 address to 2.2.2.1. For example:

```
ipcycon de0 in spoof ip 0.0.0.0:0.0.0.0 1.1.1.1 spoofaddr 2.2.2.0:255.255.255.0
```

- **Network to Network** - When the CYCON Labyrinth firewall encounters a packet coming from any source destined for any host on the 1.1.1 network, it changes the 1.1.1 address to 2.2.2 network address. For example:

```
ipcycon de0 in spoof ip 0.0.0.0:0.0.0.0 1.1.1.0:255.255.255.0 spoofaddr
2.2.2.0:255.255.255.0
```

- **Port-based Spoofing** - To add another level of complexity, the CYCON Labyrinth firewall also has the ability to distinguish traffic based on port mappings. For example, an internal web server may be used, and all incoming traffic for any local IP address with a destination port of 80 is remapped to the single web server, as follows:

```
ipcycon de0 in spoof ip 0.0.0.0:0.0.0.0 0.0.0.0:0.0.0.0 dst-eq 80 spoofaddr 1.1.1.1
```

The CYCON Labyrinth firewall also has the ability to spoof only destination ports and remap only the port. For example, an advertised web server at port 8080 and can be changed to the standard WWW port 80. The CYCON Labyrinth firewall identifies any inbound traffic destined for the internal web server on the original port and rewrites the header to map to the new destination port, as follows:

```
ipcycon de0 in spoof ip 0.0.0.0:0.0.0.0 1.1.1.1 dst-eq 8080 spoofaddr 1.1.1.1
spoofport 80
```

## IPSec - Encryption

IPSec is a set of standards for Internet security to ensure open standard host-to-host, host-to-firewall, and firewall-to-firewall connectivity. The standard includes two parts: Authentication and Encapsulation. The CYCON Labyrinth system supports these standards as specified in RFC 1825, RFC 1826, RFC 1827, RFC 1828 and RFC 1829.

The Authentication Header (AH) provides a mechanism whereby the sender signs IP packets and the receiver verifies the signature. This helps to prevent alteration of packets and spoofing during transit.

The Encapsulation Security Protocol (ESP) provides a mechanism whereby the sender encrypts IP packets and the receiver decrypts the packets. This helps to preserve confidentiality and privacy and is key to implementing virtual private networks (VPN).

### ***IPSec Filter***

The CYCON Labyrinth firewall supports IPSec as specified in the standards RFC-1825, RFC-1826 and RFC-1827. The CYCON Labyrinth firewall allows AH and ESP to pass through the system using security filter rules. AH is treated as an attribute of the protocol field while ESP is treated as a separate protocol. For example, to permit AH signed packets into interface de0, the following firewall command is used:

```
ipcycon de0 in permit ip-ah 128.33.0.0:255.255.0.0 115.27.0.0:255.255.0.0
```

The "-ah" attribute can be used on any protocol. When used, a packet must have an Authentication Header within the packet.

To permit ESP packets into interface de0, the following command is used:

```
ipcycon de0 in permit esp 128.33.0.0:255.255.0.0 115.27.0.0:255.255.0.0
```

The ESP protocol matches all encrypted packets.

These two methods only permit packets in and out of interfaces. Concurrently, the CYCON Labyrinth firewall also functions as an IPSec gateway. Using the following features, it is possible to authenticate and/or encapsulate communication to and from the firewall as well as to and from hosts on networks via the CYCON Labyrinth firewall.

### ***IPSec Gateway***

The CYCON Labyrinth firewall uses two versions of a special security key system to control the AH and ESP mechanisms within the firewall. As such, the CYCON Labyrinth firewall can be configured to sign packets (AH) on behalf of the client system, and/or check the AH signature of packets entering the network. Furthermore, the CYCON Labyrinth firewall can encrypt and decrypt communications between hosts or networks communications through the CYCON Labyrinth firewall. This is accomplished by configuring the encryption, decryption, and authentication algorithms, keys, and addresses with the spi command.

When the CYCON Labyrinth firewall is functioning as an IPSec gateway, an additional set of attributes is available for the ipcycon rules. These attributes are set for inbound rules when a packet is successfully authenticated or decrypted. Likewise, these attributes force authentication and encryption when used on outbound rules. For example, a packet decrypted by the CYCON Labyrinth firewall will match the attribute "-via\_esp". To accept decrypted packets through the de0, the following command is used:

```
ipcycon de0 in permit ip-via_esp 10.9.0.0:255.255.0.0 129.2.0.0:255.255.0.0
```

To force encryption on communications through the de0, the following command is used:

```
ipcycon de0 out permit ip-via_esp 10.9.0.0:255.255.0.0 129.2.0.0:255.255.0.0
```

Likewise, the "-via\_ah" attribute may be used to match properly authenticated packets or force authentication headers to be added to packets.

## Common Use

The most common mode of operation is to support Virtual Private Networks (VPN). In this mode, two or more LANs communicate with each other over public networks (e.g., the Internet) and maintain their security by encrypting all communications between these networks. In this mode, the CYCON Labyrinth firewall resides between the LAN and the public network. The system encrypts all traffic from the LAN before it is passed over the public network to another LAN. The system also decrypts all traffic entering the LAN from the public network. As a result, the computers on the LAN do not have to support encryption. Instead, they communicate as they would with any other system, and the CYCON Labyrinth firewall does all the work transparently to the users.

The next common mode of operations supports access to private LANs via public networks by remote users. In this mode, the remote user will use an IP stack that support the IPSec standard. If the user's IP address is dynamic, then a third-party authentication is needed to identify the user, the IP address and the encryption keys needed for the session. If the user's IP address is static, then a weaker authentication method could be used. Once the remote user is authenticated, all traffic in and out of the LAN to and from the user's address is encrypted and decrypted. This protects sensitive information from sniffer attacks while it traverses the public network.

## Protection of Attached Networks and Hosts

CYCON Labyrinth firewall intercepts, examines, decides, and either blocks or permits IP traffic passing between the protected and unprotected networks. CYCON Labyrinth firewall blocks or permits those traffic flows based on the rules that the firewall administrator creates.

Blocking and permitting of network traffic is based on CYCON Labyrinth firewall's ability to examine packet headers, compare the information against filter rules, and take an appropriate action. If the packet's header information does not directly apply to a permit rule, the packet is dropped. In addition, the stateful inspection module remembers outgoing connections and only allows the expected replies of permitted connections back through the firewall. CYCON Labyrinth firewall can perform the following actions on packets:

- **Permit** - permits the packet and routes it to the appropriate interface;
- **Deny** - denies the packet and sends an appropriate ICMP message back to the sender;
- **Drop** - drops the packet with no reply message;
- **Track** - permits the packet and creates a dynamic rule to permit expected replies;
- **Proxy** - rewrites the source address of the packet with either the address of the firewall or a range of user specified IP addresses; and,
- **Spoof** - rewrites the destination address of the packet with either the address of the firewall or a range of user specified IP addresses.

The proxy and spoof actions can redirect packets to any host on the network or on the Internet.

CYCON Labyrinth firewall protects against network spoofing with one simple rule. The filter rule will not accept packets originating from the external interface that contain source addresses that match any internal IP addresses. In addition, all source routed packets or IP fragments are dropped.

CYCON Labyrinth firewall supports standard username and password authentication and 128-bit encrypted SKEY (MD5) authentication. Inbound and outbound authentication is performed via an embedded technology called "VISA."

The firewall administrator maintains access lists of users and groups. A user must authenticate with the authentication server (which runs on the firewall, but optionally can run on a dedicated machine) before access is permitted. Upon successful authentication, the "VISA" system creates a dynamic rule permitting access for the user as defined in the access lists.

Any possible access rights are predefined by the firewall administrator and can be set to expire after a predefined time has

passed. It is possible to allow only certain types of access (e.g. Web, Telnet, ftp) to one group of users while allowing a different type of access (e.g. Archie, gopher, NFS) to another group. The "VISA" system is flexible enough to receive authentication requests from third-party servers, such as DHCP and WINNS servers.

CYCON Labyrinth firewall supports temporary and timed rules. These rules allow security policies that prevent certain protocols during specific times. An organization may want to restrict outbound Web access to non-business hours, or only during lunch time.

## Protection of Individual Hosts

No client-side modifications of software are necessary to provide host-to-firewall authentication. Inbound and outbound access can be configured to be completely transparent, require authentication for each session, or require authentication which is usable for a predefined period of duration.

As discussed above, the incorporation of IPSEC standards in CYCON's Labyrinth firewall enables the support of full featured peer-to-peer encrypted traffic by any third-party mechanism, either software or hardware. The IPSEC standards implemented provides fully compliant Virtual Private Networking (VPN) technology for net-to-net, host-to-net, and host-to-host connectivity.

## Systems Requirements

Hardware Requirements:

- Intel Pentium or Intel 486 (Pentium recommended), 100 MHz minimum for active 10 MB Ethernet, or 166 MHz minimum for 100 MB Ethernet
- 16 MB RAM minimum, 32-64 MB RAM for active Ethernet (each rule [static or dynamic] requires 128 bytes)
- 1 GB HD (IDE or EIDE) for typical sites (intensive logging requires more space and may degrade performance)
- CD-ROM (IDE recommended)
- 3.5" Floppy drive
- Mouse (recommended for initial load and setup)

## NetGuard's Guardian Firewall System - MAC Layer Stateful Inspection

NetGuard Ltd. is a software company specializing in security solutions for corporate networks attached to the Internet. The Guardian Firewall System, the company's first product, was released in 1995 and was acknowledged world-wide as a leading firewall product. The Guardian was the first firewall designed to operate on the popular Windows NT platform, and is recommended by Microsoft as a Windows NT solution .

Guardian Firewall software has won the British EMAP Networking Industry Award 1996 as "Internet Product of the Year". The judges described the Guardian as "...a sensibly thought-out package, which is easy to implement and manage...the Guardian takes a refreshing look at problems of implementing Network security..." NetGuard Ltd. is a subsidiary of LanOptics Ltd. a leading supplier of hubs and networking products. NetGuard takes full advantage of LanOptics' large customer base and field-proven experience in the network environment to provide high quality and efficient. Figure 14.13 is a screenshot of NetGuard's Web site, showing the awards and certification of this product.

### Note:

For more information, contact NetGuard Ltd, via e-mail, [info@ntguard.com](mailto:info@ntguard.com), or visit their Web site at URL <http://www.ntguard.com/>. You can also contact their headquarters at 2445 Midway Road, Carrollton, Texas 75006, Tel: (972) 738-6900 - Fax: (972) 738-6999.

According to NetGuard's press release of August, 1997 (<http://www.ntguard.com/newlan.htm>), GUARDIAN Firewall software was named "The Best of LAN Times" in the magazine's Aug. 4 review of the industry's leading Windows NT-based firewalls.

Guardian was designed to enable you to easily and accurately establish comprehensive security strategies and manage on-going corporate Internet access.

Guardian firewall is basically an Internet Control and Firewall software that protects the private network against sabotage, unauthorized information access, intrusions, and a wide range of threats initiated from the Internet. Guardian is certified by NCSA.

Guardian's firewall architecture is based on the unique MAC Layer Stateful Inspection that makes it immune to Operating System security leaks. It is available for Windows NT server and workstation operating systems.

The developers of Guardian, NetGuard, are a leading provider of advanced Internet and Intranet Security and Productivity products and is the first company worldwide to offer Internet Productivity Monitoring and Bandwidth control capabilities.

## ***A Unprecedented Internet Management Tools.***

NetGuard, besides being an effective user-friendly firewall System offers Network Administrators unique Internet Access Control Tools. Much has been said throughout this book, and everywhere in the news, about the hazards involved in connecting to the Internet, and indeed the issue of secured Internet connectivity has been the prime concern of network administrators in recent years.

The Firewall Market evolved in order to give a satisfactory answer to Internet security issues and NetGuard's Guardian, winner of the EMAP networking Industry Award under the category "Internet Product of the Year", plays a major role in setting the standards for Internet security. Thus, Guardian 2.1 ordiates and facilitates the task of Internet Connectivity Management, offering Network Administrators a variety of powerful management tools and comprehensive inside information in real-time. The following sections describe some of the most relevant features and tools provided by Guardian

### **Visual Indicator of Enterprise-Wide Agent Activity:**

Figure 14.14 shows Guardian manager screen, one of the powerful tools available with Guardian. Through it, you're able to effectively manage the firewall, including analysis of bandwidth allocated, as shown on figure 14.15 and more, as the following sections describes.

Another useful tool is the Agent Icon, as shown on figures 14.16 and 14.17, which in its minimal capacity format enables you to receive comprehensive visual indication of the overall activity by viewing on the same screen the activity of as many Agents as he/she chooses.

### **Extended Gateway Information**

Guardian also provides a comprehensive interface to gather extended gateway information through an enlarged Agent Icon, as shown on figure 14.18. As you can see on figure 14.18, the interface provides gateway Information on:

- Total bandwidth available
- Total bandwidth consumption
- Number of connections
- Number of active users, and
- Total number of users

### **Activity Monitoring Screen**

The Activity Monitoring Screen of Guardian allows auto detection of active users. Every User is represented by an icon which functions as an activity indicator, as shown on figure 14.19. The green computer screen indicates an active user and the blue one a non-active user.



## Enhanced Activity Monitoring Screen:

The Activity Monitoring Screen of Guardian, as shown on figure 14.20, can be configured to show additional user activity information if necessary, which includes:

- IP address and name (if assigned)
- Number of active connections
- Number of bytes received and sent
- Actual bandwidth allocation for each user
- Type of service in use

## Monitoring User's Connectivity

Users connectivity can be monitored with Guardian by using the connection monitoring screen, as shown on figure 14.21. By selecting a user icon on the Activity Monitoring Screen you are able to monitor a "real-time user connection monitoring" window which shows the following information about the user's active connection:

- Destination IP address and name
- Type of Service in Use
- Number of bytes received and sent
- Elapsed time for this connection
- Bandwidth allocation for each session

The Connection Monitoring Window introduces two new administrative functions:

- Allows the Network Administrator to close an active connection for a predefined period of time
- Allows the Network Administrator to create rules which determine the conditions under which a user operates.

## Firewall Strategy Wizard

The Guardian's firewall strategy wizard, as shown on figure 14.22, has two main functions:

- To assist you in creating a basic set of security strategy rules which serve as guidelines for a corporate security strategy. These guidelines rules in themselves can provide adequate security for the Network.
- Or, if you want to benefit from the more advanced features of the system, you may opt to develop a unique strategy rules using the firewall strategy editor.

Also, the Guardian Strategy Wizard has a tutorial function as well, which helps clarify the process of creating strategy rules and paves the way for independent creation of complex security strategies.

## WAN Adapter Support

Guardian can also be configured to work on WAN adapter connected to the Internet. The additional adapters on which an agent can be installed such as modems, ISDN and Frame Relay adapters can be used and installed on any NDIS compatible LAN or WAN adapter.

In many cases this new feature can eliminate the requirement to install a router, as shown on figure 14.23.

Also, as shown on figure 14.24, NetGuard has added to Guardian the capability to define several class C networks. When defining a NAT strategy, looking at the example below, two rules will be defined:

- Our network-1 - Global Network-1 ( 1st class C network)
- Our network-2 - Global Network-2 ( 2nd class C network)
- Our network-1, Our Network-2, Global Network-1, Global Network-2 are networks defined in the Network Object dialog box.

## Logoff Command on Authentication Client

While enabling authentication for users, Guardian enables requires the user to be assigned a time period to logon and work, as shown on figure 14.25.

This process is executed while setting an Authentication client, In the firewall strategy, when entering relevant Action for the user, where the total access time that the user can be logged on (Authenticate with an Access for) must be entered, as shown on figure 14.26. The interface also has a mechanism to control spoofing attacks, as shown on figure 14.27.

Figure 14.27

*Guardian enables you to specify a list of networks to be checked for spoofing attempts*

## CyberGuard's CyberGuard Firewall - Hardening the OS

CyberGuard Corporation is dedicated to providing the strongest, most comprehensive Internet, intranet and electronic commerce security solutions for organizations with enterprise-wide data networks.

CyberGuard Firewall is a multi-level secure computer that resides between internal networks - or between an internal network and the Internet - to provide a single secure connection point through which all data must travel. The firewall screens and filters all traffic to and from any public network before allowing it to pass. To eliminate the possibility of data theft or damage, unauthorized attempts to communicate with the internal network are logged and blocked.

The CyberGuard Firewall Release 3 is now expanded to run on Intel boxes. CyberGuard Firewall features a lower cost software-only entry-level option for departmental and remote office security solutions. CyberGuard Corporation claims to provide the world's most secure firewall on the Intel platform, also allowing you to integrate the firewall with industry-standard off-the-shelf hardware.

CyberGuard Firewall's Release 3 is an off-the-shelf software solution comprised of a trusted UNIX-based operating system, integrated secure networking software and a Remote Graphical User Interface (GUI) Manager.

This latest release (version 3) combines packet filtering and application proxy security in a solution that can be customized to allow two-way, incoming-only or outgoing-only communication while blocking high-risk commands. CyberGuard delivers high performance, high throughput, enterprise-wide security applications.

### Note:

For more information, contact CyberGuard Corp, at 2101 W Cypress Creek Road, Fort Lauderdale, FL 33309, Phone: 800.666.4273 or Phone: 954.973.5478 - Fax: 954.973.5160. You can also contact them via e-mail at E-mail: [info@mail.cybg.com](mailto:info@mail.cybg.com) or at the URL <http://www.cybg.com>

## The Trusted Operating System

CyberGuard's integrated suite of secure firewall components gives you the highest degree of protection against attacks. In a typical firewall solutions, if an attacker penetrates the firewall application, the unsecured operating system can be accessed and penetrated.

With the CyberGuard Firewall solution, the operating system has been hardened with extra security measures, as shown on figure 14.29, so unauthorized users or requests can not penetrate the O/S and the network. The secure operating system and secure networking software are based on multi-level security that restricts access to information based on the sensitivity of the information and the access authorization of system users.

The underlying operating system and networking software are designed for demanding security environments. The high performance operating system has the ability to process high levels of throughput without time-consuming failures.

CyberGuard Firewall technology can be utilized with your remote offices to operate secure enterprise-wide mobile security applications, secure database applications and access controls.

CyberGuard claims to be the strongest enterprise security solution available because it is built on a secure operating system that utilizes an extension of multi-level security called Multiple Virtual Secure Environments (MVSE), as shown on figure 14.29 above. MVSE matches data access to user privileges, preventing theft or unauthorized access to highly sensitive data via networks at lower levels of security.

This unique capability, Multiple Virtual Secure Environments (MVSE), allows a single physical network to be divided by security level into multiple virtual networks. Simultaneously, customers can divide their physical data servers into multiple virtual data servers, each with a unique level of security. MVSE ensures that the data at a given level of security only travels over networks at the same level of security. MVSE technology recognizes the need for protection of two separate corporate assets - the data and the network. Contemporary firewalls generally protect the network but not the data traveling across it. The CyberGuard Firewall is the only firewall to protect data at all enterprise levels.

MVSE's capacity to create over 200 virtual networks/servers from a single network/server provides the flexibility and growth potential the your company may need. CyberGuard's unique Multiple Virtual Secure Environments also provides a secure, cost-effective, multiple network implementation while extending security coverage to data traveling over the network.

## ***Intuitive Remote Graphical User Interface (GUI)***

The CyberGuard Firewall 3 on the Intel platform offers a Remote Graphical User Interface with an optional remote feature that allows you, as an administrator, to centrally control and monitor multiple CyberGuard Firewalls. This capability significantly lowers the cost of firewall administration by simplifying administration tasks and eliminating the need to have multiple firewall security administrators.

This innovative feature provides an integrated graphical environment for setup, configuration, monitoring and reporting. Based on the X Window System and OSF/Motif, the system hides internal mechanics from the user while presenting an easy-to use, intuitive interface. All features are configurable through the GUI. The online help includes window-level context-sensitive information, a table of contents, "how-to" tasks and a glossary, as shown on figure 14.30.

## ***Dynamic Stateful Rule Technology***

Security decisions made on a machine without a trusted O/S are inherently insecure. Part of CyberGuard's strong security approach is its dynamic stateful rule technology that extends common packet-filtering (as shown on figure 14.32) capabilities. Figure 14.31 shows the proxy configuration screen of CyberGuard.

CyberGuard Firewall monitors each connection to ensure that all network traffic from the client or server adheres to the network security policy and network protocol. The dynamic stateful rule technology of CyberGuard works with all IP network traffic, including UDP and ICMP and split DNS system, as shown on figure 14.33. Unlike other firewalls on the market today, CyberGuard's secure solution is not limited to TCP traffic. With dynamic stateful rule technology, CyberGuard Release 3 on the Intel Platform can identify network attacks such as IP spoofing and hijacking.

Further, CyberGuard establishes unique dynamic stateful rules for each new connection to or from the firewall, even if multiple connections are between the same client and server. The dynamic stateful rules reflect the state of the connection at any moment in time. Each connection has a unique dynamic stateful rule allowing CyberGuard to monitor the status of the individual connection and enforce its connection-specific security policy. Any packets received by the firewall that do not match are discarded as invalid, and alarms are tripped, as shown on figure 14.34. At the conclusion of each session, CyberGuard Firewall dismantles the dynamic rule to prevent hijacking of the connection.

## ***Certifiable Technology***

The CyberGuard Firewall Release 3 is designed by the same team that created the hardware/software CyberGuard Firewall solution (Release 2.2) with an operating system and integrated networking software that have been evaluated at the B1 level of trust by the National Computer Security Center (NCSC) and certified by the National Computer Security Association

(NCSA). The CyberGuard Firewall Release 2.2 was also tested in by Celar in France and is the first firewall solution to undergo ITSEC E3 evaluation in the United Kingdom.

This firewall, for Intel platform, is offered in three configurations:

- An entry-Level Option, supporting 50 users or less;
- A workgroup Option for 51-250 users; and
- An enterprise Option which has unlimited user support.

With CyberGuard Firewall 3, both Pentium and Pentium Pro processor systems (single or dual processor configurations) come with the same high throughput, scaleability and flexibility of previous versions of CyberGuard. An easy-to-use remote graphical user interface (GUI) manager allows system administrators to configure and manage the firewall from both remote and local sites. Figure 14.35 shows the basic architecture design of CyberGuard.

## ***Systems Requirements***

The following is the recommended systems requirements for configuring CyberGuard:

- Pentium and Pentium Pro processor systems (single and dual processor configurations)
- 32MB local memory
- 2 Ethernet connections (with optional additional independent connections)
- UNIX SVR4 compliant
- 2GB hard disk
- 17-inch color monitor
- 4mm DAT backup medium
- High resolution super VGA video interface
- Tower enclosure (or optional rack-mountable chassis)
- Optional encryption (U.S. and international)
- Optional WebTrack™ Internet-access tracker and controller

## **Raptor's Firewall - An application-level Architecture**

Founded in 1992, Raptor Systems is a leading company in integrated firewall security management software and services. Based on an application-level firewall architecture, the Eagle family comprises a suite of modular software components that provide real-time network security for Internet, workgroup, mobile computing, and remote office domains within the enterprise. The Eagle family, when used individually or as part of an integrated network security management system, addresses the need for network security in large and small companies. Eagle runs on Sun Microsystems, Hewlett-Packard, as well as Windows NT workstations. Figure 14.36 shows a screenshot of Raptor's Web site.

### **Note:**

For more information of Raptor's Eagle family of firewalls, contact Raptor Systems, Inc., 69 Hickory Drive, Waltham, MA 02154, telephone 800-9-EAGLE-6 or 617-487-7700, Fax: 617-487-6755. You can also reach them via email at [info@raptor.com.com](mailto:info@raptor.com.com) or on the Web at URL <http://www.raptor.com/>

# Enforcing Security at All Levels of the Network

Comprehensive security is a key strength of Raptor's Eagle family firewalls. Industry experience shows that of all attacks, the most damaging are those that rely on application data streams. Attacks at this level, as seen throughout this book, often go undetected by stateful packet filters, which only examine the protocol headers of packets at the network layer. Circuit level gateways are also vulnerable, since they lack the ability to examine application level data.

The Internet represents only one domain that must be secured. Every enterprise - whether newly emerging or an established multinational - has security needs that extend beyond unauthorized access over the public network.

Securing confidential data among and between workgroup LANs is a growing concern, see figure 14.37. An executive would never send an employee's salary review to human resources in an unsealed envelope. Nor would an engineering team leave product development plans on the table. A prospect list in unauthorized hands could mean disaster for quarterly sales. When that same data sits unprotected on a PC or within a server, however, it is susceptible to privacy breaches that would never be allowed in a "paper" world. It is common knowledge that over 85% of all computer crime is perpetrated by individuals who are authorized to use the systems they are working on. Hence, desktop PCs and workgroup LANs must be secured from "unauthorized" users within an organization as well as from Internet users.

Working at all seven layers of a network-based application gives the Eagle access to all contextual information needed to make authorization and authentication decisions, including:

- The specific type of application used
- Specific application commands and data allowed or disallowed
- The users, groups, or times of use allowed for the service
- Time and date ranges
- Authentication information

Based on this information, the Eagle makes complex security decisions. It automatically enforces service restrictions, issues alerts via email or beeper, SNMP trap or client program, and compiles a comprehensive log on all connections-whether they are allowed or not.

To derive only a portion of the information available to the Eagle, packet filtering firewalls must evaluate each IP packet individually, capturing state information on-the-fly. This makes these systems particularly vulnerable to attacks that exploit packet fragmentation and reassemble operations. The Eagle's architecture makes it invulnerable to such attacks.

Raptor defines five domains of network security to promote an integrated approach to protecting the enterprise:

- **Domain 1: Internet Security** - To protect networks exposed to unauthorized Internet access, as shown on figure 14.38, Raptor Systems offers the flagship Eagle firewall. Designed as the foundation on which any enterprise solution can be built, Eagle is a flexible, application-level firewall that secures bi-directional communications through the public network. It includes EagleConnect virtual private networking, a powerful, real-time network security management facility with intuitive GUI, suspicious activity and alert monitoring, encryption and multiple types of authentication and proxy software to foil IP spoofing attacks. Multiple hardware platforms are supported including Sun Microsystems, Hewlett Packard and Windows NT on Intel and DEC Alpha platforms.
- **Domain 2: Workgroup Security** - Raptor Systems provides two solutions to protect sensitive data that reside at a workgroup level, as shown on figure 14.39. The EagleLAN is a departmental firewall that integrates seamlessly with the Eagle. If one department attempts to access another department's data without authorization, the network administrator will know immediately. As with our Eagle firewall, real-time alarms let administrators catch hackers in the act. And for desktop security, EagleDesk resides on a user's PC, behind the firewall, to provide secure communications between the PC and any other authorized destination inside or outside the enterprise.
- **Domain 3: Mobile User Security** - The combination of portable PCs, telecommuters, and virtual offices opens the door to data access anywhere in the enterprise from anywhere in the world through public and private networks. To protect this newly-emerging mobile portion of the enterprise, Raptor Systems provides EagleMobile (see figure 14.40). An option to the Eagle firewall, EagleMobile can be installed by a non-technical user on any portable or off-site PC for additional password protection and encryption between their PC and an Eagle firewall.
- **Domain 4: Remote Site Security** - To secure communications among corporate headquarters, corporate divisions, and

branch offices (see figure 14.41), Raptor offers the EagleRemote firewall. EagleRemote includes all of the superior security features of the flagship Eagle firewall for remote sites that must use the public network to communicate with other enterprise "satellites." The EagleRemote is configured and monitored by the Eagle firewall. This allows the network administrator to have complete control from one central location back at the enterprise.

- **Domain 5: Integrated Enterprise Security** - As shown on figure 14.42, Raptor has designed its products as a suite of modular software components that can interact seamlessly with each other using a common management and monitoring capability. This building-block approach to security management lets companies change and grow their network security systems without changing their underlying security strategy. Central to this integration is Raptor's EagleConnect virtual private networking technology, which transparently manages the connections among network security points within the enterprise.

Eagle's strong, rules-based defense (see screenshot on figure 14.43) is very impressive. Packet filtering firewalls authorize passage of IP packets on a first fit rule matching basis. As packets enter a router or filtering firewall, the device compares each packet in turn against a set of match conditions (filters).

By default, the device accepts the first fit to these conditions to allow or deny the packet. Herein lies the problem: filtering rules are inherently general and highly order dependent. This means that the first match triggered may allow a connection that would be denied by a subsequent comparisons. Thus, whether a packet gets into your network may depend on the way you order of the rules, rather than on the rules themselves. This complexity makes misconfiguration an ever present possibility.

Therefore, with the Eagle Firewall,

- All connections are denied unless explicitly permitted
- Automatic suspicious activity monitoring
- Comprehensive logging for all connections
- Fine grained access-controls and service restrictions
- "Best-fit" Rule management

The Eagle's best fit approach is simpler, tougher, and easier to manage. To begin with, the Eagle denies all network traffic except for that which is explicitly allowed. Second, the rules the Eagle applies are not order-dependent, so it always chooses a rule specific to the connection attempt at hand. And to make sure the rule chosen is specific, the Eagle always applies conservative best fit criteria to allow or deny a connection. And if no rule meets its best fit criterion, the Eagle denies the connection. This approach to rule management by the Eagle firewall allows a firewall administrator to concentrate on the creation and management of a security policy rather than on the management of the firewall itself.

## ***Reliance on Dedicated Security Proxies***

The Eagle uses secure application proxies to examine each attempt to pass data in or out of your network. As discussed throughout this book, proxying connections provides the strongest safeguard against network intrusion. These proxies provide:

- Protection against application level attacks
- Automatic hiding of all internal IP addresses and their associated systems
- Strong and weak user authentication
- Comprehensive logging of all activity
- Fine grain control of direction of service, e.g. FTP put versus get.

The Eagle's secure proxy architecture presents a virtual brick wall between your networks and the unsecured world of the Internet. This wall protects you in two ways:

1. Only connections explicitly allowed are permitted. This greatly simplifies configuration. This in turn virtually eliminates security breaches arising from mismanagement.
2. Your networks are not only protected but hidden from the outside world. This bars hackers from probing for insecurities in your internal systems, and safeguards the critical information needed to mount an attack.

# ***Using Raptor's Firewalls Eagle Family***

Eagle is very easy to set up and use. Its richness of function and flexibility are married to a graphical user interface that makes configuration and monitoring easy.

## **Graphical Policy Configuration**

The Hawk graphical user interface segregates all aspect of your security set-up into discrete areas of function. You use one window to write rules, and others to define internal and external systems, specify firewall users, create authentication templates, and perform other functions. This makes the process of rule authoring straightforward, and the rules you write easy to understand. The Eagle's monitoring window gives you a birds-eye view of all connection attempts into your network. Its log file window displays statistical information on all connections at a glance.

According to Raptor, the Eagle is the only product in the industry with graphically configurable service proxies for all key services, including:

- HTTP (Web browsing)
- SMTP
- TELNET
- GOPHER
- SNMP (due on next release, by mid-November of 1997)
- FTP puts and gets (file transfer)
- DNS (name resolution)
- RealAudio
- Secure Remote Login (remote management)

In addition to supporting commonly used applications with out-of-the-box proxies, Hawk makes it a snap to specify additional applications.

## **Consistent Management- Locally or Remote**

Whether you are managing the Eagle locally, or via an encrypted Internet link, Hawk presents you with the same management interface. So there is never any doubt about whether the policies you put in place are really in force.

One of the key requirements for administrators is to be able to easily and securely gain access to the host operating system that the Eagle runs on top of. Raptor provides a Secure Remote Login (SRL) capability that allows administrators to remotely gain access to the operating system for configuration and maintenance. SRL establishes an encrypted and authenticated TELNET session to the firewall system.

The Eagle allows you to enforce policy decisions for end users, while making it easy for them to get their jobs done. Whether this entails use of Web browsers, file transfer, or remote login to selected systems, the Eagle's presence is unobtrusive. In most cases, users are not even aware of the Eagle's operations.

## **The Flexibility to Allow "Transparent" Access**

While it presents an unbreachable wall to unwanted users, the Eagle provides flexible access to users you need to accommodate. In fact, you can configure the Eagle so that users will not be aware of its presence.

Usually referred to as transparency, this level of access allows users to "see" and (apparently) connect directly to certain systems. These connections are still proxied by the Eagle, which continues to carry on extensive logging and alerting operations. So even though your users may be unaware of it, the Eagle is still watching the store.

## Address Redirection

At times, you may need to allow users to access data on certain internal systems, and still conceal these systems' identities and addresses, as showing on figure 14.47. Examples of this could include customer information databases or commerce servers: resources that you must both protect, and provide access to from the outside world. The Eagle can be configured to present one or many public IP addresses which can then be mapped or redirected (on a per service basis) to systems behind the firewall with different (and hidden) IP addresses. A common use is to map multiple public IP addresses to multiple and different Web servers behind the firewall.

As for performance, independent lab tests performed at the National Software Testing Laboratories (NSTL) confirm the Eagle as the fastest transaction processing engine of any tested.

The Eagle's application proxy architecture is the key to its great performance. Since the Eagle authorizes connections at the application-level, it has access to all contextual information on each connection attempt. As a result, the Eagle only needs to evaluate each connection once. No additional checking is needed to proxy packets securely. This delivers a big performance advantage over other approaches.

## Fine-grained control of VPN Tunnels

The ability to apply packet filters within configurable Virtual Private Networking (VPN) tunnels, as shown on figure 14.44, provides Eagle administrators with fine-grained control of the types and direction of traffic that can be passed between hosts or systems. This control boosts overall network performance by enabling you to specify appropriate levels of encryption for each tunneled application.

The Eagle performs all filtering on the VPN tunnels you establish between trusted systems. All traffic passed between these systems is encapsulated and encrypted by cooperating Eagle systems, as shown on figure 14.46. This ensures the privacy and integrity of the communication. The additional use of packet filters provides an even higher level of security on these trusted tunnels, allowing only certain types of traffic in specifiable directions. Figure 14.45 shows the monitoring of real-time suspicious activity of Raptor's Eagle family firewall.

## Integrated Web Blocking Capability

The Eagle's integrated WebNOT software gives you the ability to restrict web browsing from sites containing objectionable materials. The service restrictions the Eagle supports give you the power to limit browsing activities in specific, carefully defined ways. This ensures that your organization gets the full benefit of the Internet's resources, while avoiding the unnecessary risks and performance degradation.

**Tip:**

For more information on WebNOT, check Raptor's URL at <http://www.raptor.com/products/webnot/webnot.htm>.

## *HTTP Service limitations*

In addition to the WebNOT blocker, the Eagle gives you the tools you need to limit Web access and content retrieval. Controls available for HTTP rules include:

- Filtering of designated MIME types, including Java applets
- Filtering of file types by extension
- Filtering by designated URL
- Automatic filtering of specific HTTP attacks related to buffer overruns, embedded 8-bit characters and illegal URL formats



# Systems Requirements

Raptor's UNIX firewall is available on Sun Solaris and HP-UX. Now in its fourth generation, Eagle NT provides the same robust security and flexibility of our award winning UNIX variant, tightly integrated with the Microsoft Windows NT platform.

The Eagle supports the broadest range of authentication types in the industry. It's design makes it easy to combine weak forms of authentication (like gateway password and NT domain) and strong, single-use password schemes in a single rule.

According to Raptor, the Eagle firewall family is also the first commercially available firewall to offer full support for IPSec, including DES, triple DES and RC2 encryption. Additional standards supported include SNMP V1 and V2 traps, and NT Domain, TACACS+ and Radius authentication types.

## Milkyway's SecurIT FIREWALL - a Factory Hardened BSDI Kernel

Milkyway Networks, incorporated in 1994, is a leading global supplier of Internet and Intranet security applications designed to safeguard corporate-wide information. The company's vision is to provide a single security solution for internetworking, no matter where users or servers are located on the network.

SecurIT FIREWALL is the centerpiece of the Milkyway SecurIT SUITE, the industry's first bundled suite of security products that leverages the power of Milkyway's flagship Black Hole technology with a secure, remote access product and a network security auditing tool. Milkyway's firewall product has been evaluated by the Canadian Security Establishment as an information security product achieving international draft functional specifications and "tested and certified" by the National Computer Security Association in the US. It has also been identified by Network World as the most innovative firewall. Milkyway is the first firewall vendor to incorporate a "factory hardened" UNIX kernel, which experts agree is more secure than other approaches that merely filter out unauthorized Internet addresses or use unhardened operating systems. Figure 14.47 shows a screenshot of Milkyway's Web site.

### Note:

For more information, contact Milkyway Networks Corp., 2650 Queensview Drive, Suite 150, Ottawa, ON - CANADA, K2B 8H6 or via their distributor in U.S., North Eastern, 109 Danbury Road, Office #4B, Ridgefield, CT, USA, 06877. By telephone, dial 613) 596-5549 or 800) 206-0922, Fax: (613) 596-5615 or via e-mail at [info@milkyway.com](mailto:info@milkyway.com) and Web site at URL <http://www.milkyway.com/>

## A Bullet Proof FIREWALL

SecurIT FIREWALL acts like a security guard to protect your private network from the Internet, as seen on figure 14.49. But the people at Milkyway know that the security guard itself must be protected to remain effective. Protection is crucial, you do not want your security guard to be attacked while on duty.

To protect the firewall, the SecurIT FIREWALL kernel has been "hardened" to eliminate insecure processes. Thus the firewall is very secure and will stand up to any attack. In fact, SecurIT FIREWALL also monitors for many types of attacks and alerts the system administrator if an attack is in progress. Figure 14.50 illustrates how SecurIT FIREWALL controls and monitors network visibility.

## Building a Secure Kernel

For any operating system, the Kernel is responsible for resource allocation, low-level hardware interfaces, and security. The configuration of the kernel dictates the functions that the operating system supports and includes everything from basic functions like hard drive access and video support, to more advanced features such as sound card support. To enhance security, Milkyway also suggests a dual SecurIT FIREWALL setup, as illustrated on figure 14.51, but the secure kernel is one of the main features of SecurIT FIREWALL.

### Note:

#### Configuring a Dual SecurIT FIREWALL

The following policy is used in this configuration:

- Inside Network users can access the Private Network transparently:
- Inside Network users can have an Inside DNS/Mail server or they can access the DNS/Mail server on the Private Network. Similarly, Inside Network users can have an Inside News server or they can access the News server on the Private Network.
- Private Network users will need user level authentication to access the Inside Network.
- Private Network users and Inside Network users can access the Internet transparently (or they may need user-level authentication for going through the Outside SecurIT FIREWALL, if so configured by the system administrator).
- Internet users will need user-level authentication to access the Private Network.
- Internet users CANNOT access the Inside Network.
- This policy is a combination of a rule on the Inside SecurIT FIREWALL and a user-base security policy.
- This policy combination requires that an authorized user from the outside, after having connected to a machine on the Private Network, CANNOT start a session on that machine to another on the Inside Network (even if the user is normally allowed to do so from within the Private Network). Since users who have access to the Inside Network are considered trusted, this policy should not be difficult to enforce. Otherwise, do not allow any incoming sessions to the Inside Network.

In this configuration, all the internal users on both the Private Network and the Inside Network still enjoy transparent access and the Inside Network is immune to access to the Internet by a man-in-the-middle attack.

The Dual SecurIT FIREWALL configuration provides the ultimate defense against man-in-the-middle attacks to the protected sub-network and allows all users (private and sub-net) transparent access to the internet.

To build a secure kernel for SecurIT FIREWALL, Milkyway started with a standard UNIX kernel for the platform on which SecurIT FIREWALL was to run (a Sun Sparc kernel and a BSDI kernel). Then the kernel was modified to remove all non-essential functions, resulting in a kernel that only supported TCP/IP networking, hard drive access, and similar basic functions on a restricted selection of platforms. The result is a specialized and very secure kernel but with limited functionality.

Functionality was carefully added to support the needs of a firewall. Care was taken to ensure that all functionality that was added was secure. The resulting SecurIT FIREWALL kernel is a very secure hardened kernel that has limited and specialized functionality. In addition, the kernel has also been made untouchable so that it cannot be accidentally modified (and its security compromised) by the administrator.

This limited functionality means that the SecurIT FIREWALL kernel does not support a wide range of devices but support is limited to devices essential to a firewall. As new devices are developed, before they can be supported by the SecurIT

FIREWALL kernel they must be evaluated by Milkyway and only added if they are essential and a secure way can be found to support them.

For this reason SecurIT FIREWALL does not support all types of network cards. In fact, support for two network cards was not added to the kernel because the vendors of the cards could not supply Milkyway with drivers that would allow secure support of the product.

## ***SecurIT FIREWALL Kernel Modifications***

When designing SecurIT FIREWALL, Milkyway examined the standard kernel and identified seven network functions that can cause security vulnerabilities. To protect against these vulnerabilities, the SecurIT FIREWALL kernel:

- Disables automatic source routing so that the firewall does not route any packets automatically. All packets that are received by the firewall must be authenticated.
- Disables Internet Control Message Protocol (ICMP) redirect functions. If enabled, these functions allow remote users to change routing. Disabling ICMP redirect protects SecurIT FIREWALL from this sort of tampering.
- Disables IP forwarding so that the firewall does not act as a router. All TCP and UDP packets are forced to be processed at the application layer rather than the kernel layer, where the packets can be authenticated.
- Disables communications on the syslog ports. The syslog ports are used by the SecurIT FIREWALL system log and disabling communication on these ports protects the firewall system log from being altered.
- Monitors all 64,000 TCP/UDP ports to detect all connection attempts. No connection is possible on any port until it is authenticated. No other firewall is able to monitor all ports.
- Verifies IP packet direction to eliminate the possibility of an intruder on the Internet masquerading as an internal IP source address. This firewall also verifies the direction of the traffic flow to detect and log all IP spoofing, and masquerading attempts. Milkyway's firewall also verifies packet direction for all interfaces to the firewall, not just the interface to the Internet (called the insecure interface).
- IP packet absorber functionality has been added, so that the network layer accepts any packets received on any of its configured devices. All packets are forwarded to the kernel layers above the network layer. This permits SecurIT firewall to spoof the originating host into believing that Black Hole is the actual destination machine.

## ***Kernel Security Features are Certified By CSE***

According to Milkyway, SecurIT FIREWALL successfully completed an EAL-3 Common Criteria (CC) evaluation from the Communications Security Established (CSE).

The CSE, a Canadian federal government organization, evaluates commercially available information security products under the Trusted Product Evaluation Program (TPEP) to ensure that such products meet stated functional specifications. Thus, the Canadian Security Establishment (CSE) has certified that the Black Hole technology, including the basic secure kernel and all of the additions made to the kernel, function as documented.

## ***Key Management***

Key management is one of the most difficult and crucial aspects of providing a usable and trusted virtual private network. The basic problem is how to provide all trusted users with access to up-to-date keys while keeping private keys from being intercepted by people outside the realm of trust.

SecurIT FIREWALL uses the Entrust Public Key Infrastructure (PKI) as a mechanism for authentication and encryption using public keys. This PKI is based on the X.509 standard for authentication and encryption.

Automated key distribution using Nortel Entrust PKI means that once identity is established, distribution of public keys is managed automatically. Key distribution using an X.500 database and Version 3 X.509 certificates can be centrally managed by a third-party key management service or by an in-house key management system.

Automated key distribution provides all SecurIT FIREWALLs on the VPN with easy access to up-to-date public keys for any other SecurIT FIREWALL on the VPN.

## ***Key Management and Certification Service***

A third-party key management service, such as Stentor's OnWatch, uses an Entrust/Server to create an identity for each node of your VPN. The identity includes public keys, which are stored in the key management service's X.500 public key database. Figure 14.52 illustrates this concept.

When two SecurIT FIREWALLs use Entrust/Session to start a VPN session, they authenticate each other using SPKM. The key management service is also the certificate authority for authentication of the public key. The advantages of using a key management service are the ability to provide the best security possible with a minimum of administration.

## ***In-house Key Management***

In-house key management involves creating an X.500 database behind one of the SecurIT FIREWALLs on the VPN. Entrust/Server can be used to create identities and manage public keys in the X.500 database, as shown on figure 14.53. In-house key management can provide virtually the same quality of security (key management and certificate authority) as using a key management service. But keep in mind the operating cost of this, as running Entrust/Server in-house and maintaining an X.500 database is usually an option for larger organizations.

## ***Manual Public Key Management***

The key management and distribution systems described previously employ Entrust/Session running on SecurIT FIREWALL and Entrust/Server to provide key management. A third option is to use Entrust/Lite to provide key management and create public and private keys for each SecurIT FIREWALL on the VPN, as shown on figure 14.54.

Entrust/Lite incorporates the standard Entrust features, except that Entrust/Lite does not require an X.500 infrastructure and does not support automated key distribution. Instead, Entrust/Lite creates an address book containing public keys for each SecurIT FIREWALL on a VPN. This address book must be distributed to each SecurIT FIREWALL, and each copy of the address book must be kept up-to-date.

## ***Private Keys***

SecurIT FIREWALL supports the use of private keys for data encryption and decryption, as shown on figure 14.55. Note that while a private key system requires very little overhead, it may be difficult to keep private keys for many SecurIT FIREWALLs up-to-date in a reliable and secure manner.

## ***Something Else You Should Know: Ubiquitous Monitoring of All Ports***

As mentioned in the section above, SecurIT FIREWALL is the only firewall capable of listening ubiquitously to all ports to detect and report any attempt to communicate with the firewall. SecurIT can intercept any attempt by an intruder trying to gain access to the firewall or the private network being protected by the firewall. When an intruder is detected, SecurIT logs all of the details of the intrusion attempt and alerts the system administrator.

Securely implementing Internet access, Intranets and Extranets is as confusing as ever with a myriad of security technologies, claims and concerns to consider. While "crackers" account for the vast majority of external intrusion attempts, internal incidences account for 70% of all security compromises. Industrial espionage is the most serious threat to a company, though it accounts for a very small portion of detected problems. Therefore, a layered "belt and suspenders" approach is essential for protecting your organization's networked assets. Figure 14.56 shows the fundamental components of corporate security as seeing by Milkyway, as the base for their firewall product development.

## ***Watch for Port Numbers: The Milkyway Way***

For a packet of information to be received by a computer communicating across the Internet, the packet must include a port number. The port number identifies the network service required to receive the packet. For example, if a computer is running an FTP network application, it can receive packets containing the FTP port number. If no FTP network application is running, the computer cannot receive FTP packets.

All network applications are assigned a port number. FTP uses port 21. Telnet uses port 23 and so on. There are a total of 64,000 ports. The port number is used by a computer receiving a packet to determine what application or service is required for the packet. If there is a network service running that can receive the packet, the computer can receive information on that port. If the network service is not running, then the computer does not receive information on that port.

A common first step to gaining access to a computer is to run a port scanning program against the computer. The port scanner attempts to communicate with the computer using each communications port and reports back the ports that receive information.

Knowing which ports receive information lets an intruder know what network services can be used to access the computer.

For example, if the port scanner found that the computer was accepting packets sent to port 21, this means that the computer is capable of communicating using FTP. This allows the intruder to attempt to use an FTP program to access the computer or to exploit known FTP weaknesses.

One of the strongest feature I find on SecurIT is that it listens on all ports. Listening on all ports means that this firewall accepts communications on all 64,000 ports, which has two important consequences:

- All ports accept communications
- All attempts to connect to the firewall are intercepted.

As far as I can tell, as I write this section (August 1997), listening on all ports is unique to SecurIT firewall. This is a very important feature, as an effective way to protect a system from unauthorized access is to prevent an intruder from learning anything about the system. As discussed earlier, port scanning normally provides an intruder with exploitable information about a system. However, if all the hacker learns is that all ports are accepting communications he/she is no further ahead. There is nothing to distinguish one port from another. No new information is gained.

Further, any attempt to connect to any port on a SecurIT firewall is recorded by the Logging Facility. The information logged includes the source address of the connection attempt. This information can then potentially be used to determine the source of the attack.

In addition, the Alarm Facility of this firewall continuously analyses logging information and will raise an alarm if compromising activity (such as port scanning) is recognized.

## Defending Against Common Attack Methods

As discussed earlier, listening on all ports protects SecurIT FIREWALL, and the networks behind SecurIT FIREWALL, from most attacks. In addition to the broad-band protection offered by listening in all ports, SecurIT FIREWALL has other security features built in to protect against other kinds of attempts to gain unauthorized access.

### ***Buffer Overflow***

A buffer overflow occurs when a program adds data to a memory buffer (holding area) faster than it can be processed. The overflow may occur due to a mismatch in the processing rates of the producing and consuming processes, or because the buffer is simply too small to hold all the data that must accumulate before some of it can be processed.

Software can be protected from buffer overflows through careful programming, but if a way to cause a buffer overflow is found, the computer running the software can be compromised. If a user accesses a computer across the Internet and intentionally causes a buffer overflow, the program that the user was running may crash but the user may remain connected to the computer. Now, instead of accessing the computer through the controlled environment of the program, the user may have direct unrestricted access to all of the data on the computer.

Milkyway codes the programs (for example, proxies) that run on SecurIT FIREWALL to stop buffer overflow from occurring. Even if a buffer overflow occurs, the proxy crashes because the memory "box" in which the proxy runs is protected from buffer overflow. Also, when the proxy crashes the user is disconnected because the connection depends on the proxy.

In addition, protecting the memory buffer means that the firewall keeps running and security is not compromised. If a

firewall that is not protected in this way encounters a buffer overflow the entire firewall may crash, causing a service disruption.

## ***Trojan Horses Running on the FIREWALL***

If you remember, a Trojan horse is a program designed to break security or damage a system but that is disguised as something benign. There is no way to load or run unauthorized applications on SecurIT FIREWALL. Thus a program used to create a Trojan horse would not be able to run.

### ***Spoofing***

Spoofing can occur when a packet is made to look like it came from an internal network even though it came from an external one. SecurIT FIREWALL eliminates spoofing by recognizing the firewall interface that specific source addresses can connect to. If a port receives a packet that should only be received at another port, the packet is denied.

### ***Sniffing***

Sniffing involves observing and gathering compromising information about network traffic in a passive way. This can be done by any node on a non-switched Ethernet. On non-broadcast media (for example, ATM, T1, 56k, ISDN) an intruder would either have to be in the telephone switches, have physical taps, or easiest, break into any router where the data travels.

SecurIT FIREWALL does not prevent people from sniffing the external network. As a matter of fact, no firewall can prevent that! However, since the firewall keeps external people from breaking into the internal network, this effectively prevents external people from running sniffers on the internal network.

### ***Hijacking***

Hijacking a connection involves predicting the next packet in a TCP communications session between two other parties and replacing it with your own packet. For example, hijacking could be used by an intruder to insert a command into a Telnet session. To hijack successfully, an intruder must either make an educated guess about the TCP sequence information, or be able to sniff the packet.

Hijacking is a threat because the intruder can wait for users to authenticate themselves, and then the intruder can take over the authenticated connection. Hijacking of a connection can happen no matter how strong the authentication required to start the connection

Since traffic on the networks protected by SecurIT FIREWALL cannot be seen, and cannot be sniffed, this firewall prevents hijacking attacks on traffic that does not pass through the firewall. Figure 14.57 shows Milkyway's product family at glance to protect against all the issues discussed in this section. Figure 14.58 shows a screenshot of Milkyway's site at URL <http://www.milkyway.com/prod/info.html>, which provides a product information matrix. I recommend you to access this page for additional information.

## ***Systems Requirements***

The following is the recommended systems requirements for configuring SecurIT:

- Pentium and Pentium Pro processor systems (single and dual processor configurations)
- 32MB local memory
- 2 Ethernet connections (with optional additional independent connections)
- UNIX SVR4 compliant
- 2GB hard disk
- 17-inch color monitor
- 4mm DAT backup medium
- High resolution super VGA video interface

- Tower enclosure (or optional rack-mountable chassis)

# WatchGuard Technologies's Watchguard Firebox System - Combining All Major Firewall Approaches into a Firebox

Founded in 1996 and based in Seattle, Washington, WatchGuard Technologies' founders and engineers have expertise in network management and firewall technology from previous entrepreneurial ventures, including the highly successful Networx and Mazama Software Labs. WatchGuard Technologies is building on this heritage by delivering next-generation Internet/intranet security products that eliminate the cost and complexity associated with current offerings and feature powerful hybrid firewall technology plus intelligent security management at an affordable price. During late summer of 1997, the company unveiled WatchGuard SchoolMate, the first firewall product intended specifically for use in schools. Based on the low-cost, plug-in-appliance designed for mid-sized corporations, it integrates Microsystems's CyberPatrol filtering software with the WatchGuard Firebox system.

According to WatchGuard Technologies, their firewall product, Watchguard Security System, is the industry's first network security appliance and Windows-based security management system. It is also the industry's lowest cost complete firewall solution, and the first to bring high function firewall protection to Microsoft network administrators without extensive UNIX networking expertise.

The WatchGuard Security System is also the first product to Figure 14.59 shows a screenshot of WatchGuard Technologies Web site.

## Note:

For more information, contact WatchGuard Technologies Labs, Inc. at 316 Occidental Avenue South, Suite 300, Seattle, WA 98104. Tel.: 206/521-8340 and Fax: 206/521-8341. Or you can visit their Web site at URL <http://www.sealabs.com>

## WatchGuard at Glance

WatchGuard offers you all major approaches to firewall design, such as packet filtering, proxies and stateful inspection as many of its competitors, however, with a low cost and easy to use interface. It also adds features not easily available in other similar products, such as inspection of executable content such as Java and ActiveX and the ability to e-mail you with traceroute and finger information.

Basically, the WatchGuard System consists of the WatchGuard Firebox, a network security appliance featuring a Pentium processor, and WatchGuard Security Management System (SMS), software that runs on Windows NT, Windows 95 and Linux workstations.

The WatchGuard "point-and-click" approach makes it very easy to install and configure the firewall. Configuration information is presented on a service-by service basis, allowing you to setup security even if you don't have extensive knowledge of your network. You only add Internet services you wish to enable, keeping access to a minimum and security to a maximum. Also, WatchGuard's visualization tools allow you to get a complete picture of your network security and see overall trends and network usage patterns.

WatchGuard has the ability to automatically warn you of security-related events occurring at the firewall. It delivers these messages by e-mail, pager, or custom script to almost any device, computer, or program that you use. It can provide detailed logging of every firewall event or simply record events that you designate to be significant. Thus, you can test for "holes" and see at-a-glance what visitors to your site can and cannot do.

The Firebox itself is a dedicated network security "appliance". It contains a real-time firewall operating system giving you the ability to be up-and-running right out of the box. The firewall operating system does not allow user log-ins and only supports encrypted connections to the Firebox from the SMS software.

As a standalone element, the security appliance is a specialized solution. As such, the WatchGuard Firebox is more reliable than a general-purpose system modified to do the specialized work of network security.

Other advantages associated with the standalone, dedicated nature of the appliance include the following:

- It plugs into the network and is operational within minutes. As a dedicated device rather than a general-purpose computer, it is simpler to boot up and run.
- It is managed from an ordinary desktop Windows 95 or NT PC that is used for other functions, yet it serves any network-PC, Macintosh, or a cross-platform environment.
- Its specific configuration makes it easier to verify security performance. In a general-purpose OS, a stew of network drivers, devices, and third-party software produces unbounded and sometimes undetectable security risks.
- Its exclusive focus on security ensures that it does not degrade the router or the network server's performance.

WatchGuard is built around the basic premise that unless an external user has authorization for a specific activity, then that external user is denied an inbound connection. The second premise is WatchGuard's ability to enforce security even if your network fails. It ensures that your site and the SMS software itself are not under attack by intruders. If WatchGuard suspects that its own software has been tampered with, it shuts off access to your network before an intruder can circumvent its protective screen.

## ***WatchGuard Security Management System***

As illustrated on figure 14.60, WatchGuard consists of two major components, the Security Management System (software) and the Firebox (hardware). The Security Management System (SMS), as shown on figure 14.59a, configures and monitors the Firebox and performs logging and notification of firewall events. SMS provides a secure gateway or firewall between any combination of IP hosts and IP networks. It can act in the following ways:

- **InternetGuard**, to protect corporate networks and bastion hosts from the Internet and to define company-level security
- **GroupGuard**, to protect departmental systems, restrict information and packet flow and define group-level Internet privileges
- **HostGuard**, to protect mission-critical servers with crucial databases

WatchGuard's Security Management System runs on standard Windows 95, Windows NT or Linux workstations that can be connected to the WatchGuard Firebox over a LAN or directly via a serial cable connection. WatchGuard SMS software includes all firewall setup and configuration software as well as the WatchGuard graphical user interface which is based on a service-centric model, meaning that you add only the services that you wish to enable, keeping access to a minimum and security to a maximum.

The WatchGuard SMS includes a powerful alarm and event notification system that serves to alert you to attempted security attacks while automatically blocking scans. It also includes a "reverse probe" capability that traces scan attempts back to the originating host address.

With the event notification system, network managers can choose to be notified of attempted break-ins either via email or pager messages. They also can establish a threshold number of attempts to set off the alarm system in order to avoid being "flooded" with messages.

The WatchGuard graphical interface, as you can see on figure 14.61, is based on a service-centric model, meaning that you add only the services that you wish to enable, keeping access to a minimum and security at a maximum. WatchGuard's operating system has been "hardened", as many other products reviewed on previous sections, which helps to eliminate security holes and ensures reliability.

The following is an itemized list of features available with WatchGuard:

- Block unwanted traffic into and out of the network
- Camouflage of internal host IP addresses from outside network
- Inspect e-mail for likely hacker commands
- Control FTP privileges



- Inspect Web traffic for dangerous mime types (i.e. Java, ActiveX, PostScript, etc.)
- Notification system alerts you to attacks and scams
- Visually depict traffic and usage
- Optional add-on modules

## ***WatchGuard's Firebox***

As mentioned earlier in this section, WatchGuard consists of two major components, the Firebox (hardware) and the Security Management System (software). The WatchGuard Firebox is a hardware firewall platform that runs the transparent proxies and the dynamic stateful packet filter to control the flow of IP information.

The WatchGuard Firebox resides between your router and your trusted local network, which connects to local workstations and servers. The Firebox also provides an interface for an optional bastion network which might contain servers (for FTP and World Wide Web for example) you wish to be accessible from the Internet with different access policies than the machines on your trusted local network.

The Firebox is a specially designed, properly optimized machine for running the WatchGuard firewall. It is designed to be small, efficient and reliable, as seen on figure 14.62.

The following is an itemized list of the Firebox features:

- Real-time embedded operating system
- Stream-lined firewall engine
- Camouflages internal addresses
- Tamper-proof operation
- Inspects and blocks unwanted traffic

## ***WatchGuard's Global Console***

As showing on figure 14.63, the WatchGuard Global Console depicts real-time status of each firewall on the network. It gives network administrators the ability to easily manage multiple firewalls from a single location.

Essential information about each Firebox, such as contacts, phone numbers, IP addresses, and configuration information, is organized and accessible for each Firebox making on-the-fly configuration and monitoring quick and easy.

For management ease, an overview of the real-time status of all Fireboxes on the Internet is summarized on one screen. Easy-to-understand icons indicate various firewall states including whether or not the system is running, the amount of traffic over the firewall, or if a packet has been denied. The console also generates in-depth details about each state, as illustrated on figure 14.64.

The following is an itemized list of the main features available in the Global Console:

- Real-time status of all Fireboxes summarized on one screen
- Easy-to-understand icons
- Configure any Firebox from a single location
- Critical and important information organized for easy access for each Firebox
- Encrypted session links to multiple Fireboxes
- Easy zoom in to detailed information for each individual Firebox with standard SMS tools.

# WatchGuard Graphical Monitor

The WatchGuard Graphical Monitor is the perfect complement to the WatchGuard Security Management System (SMS). It is composed of three separate programs that monitor three different aspects of your network.

- **HostWatch**, as showing on figure 14.65, shows real-time graphical representations of host to host activity on your network, allowing you to watch as connections begin and end. Arrows indicate the direction of the connection and the type of service is indicated by icons, displaying at a glance the type of connection that is occurring between hosts. HostWatch also allows instant replay of activity based on your log files. This allows you to review your network's activity at your leisure, or look for patterns over several days or months.
- **ServiceWatch**, as shown on figure 14.66, plots the number of connections occurring for a specific service so that you can monitor the composition of your network traffic.

The Mazameter monitors the amount of bandwidth being used by your network. It can graph usage on scales, as shown on figure 14.67, from dial-up to full T1 to identify when your Internet connection is busiest.

## WatchGuard Reporting System

Tired of searching through logs and writing custom scripts to sort and tally your network usage? The WatchGuard Historical Reporting Module, as seen on figure 14.68, provides an easy interface which gives you a quick summary of network activity, as well as the ability to export the information to any database.

Configurable searching based on time spans, clients, and services is available with the WatchGuard Historical Reporting Module. As you can see on figure 14.69, standard reports include top ten clients, top ten services, incoming connections based on time of day, outgoing sessions for a particular client during a particular time, and many more.

## WatchGuard WebBlocker

WatchGuard WebBlocker is a tool that provides tailored management control over web surfing putting Web site access privileges fully under the control of corporate managers. Because WebBlocker is flexible, users can block all browsing of the Web by user group and times of day.

For example, corporate managers can use WebBlocker to prevent selected departments and work groups from accessing all of the selected site categories (see figure 14.70) during normal business hours, but allow access to categories such as sports and leisure during lunch breaks and after 5:00pm. WebBlocker also provides users the ability to add the names of sites they wish to permanently block or permit, as shown on figure 14.70a, in keeping with their corporate access requirements.

WatchGuard WebBlocker is based on Microsystems Software's Cyber Patrol database. Each week automated updating of the WebBlocker database is downloaded via a secure, encrypted Internet connection. The list of supported groups feature questionable or inappropriate content.

WebBlocker set-up software vastly simplifies the creation of customized group profiles as well as other configuration tasks, as shown on figure 14.70b. The WebBlocker set-up walks users through each step of the process and lets them map different access privileges to different groups using simple point-and-click operations.

## WatchGuard SchoolMate

As I write this section, WatchGuard SchoolMate stands as the first firewall product intended specifically for use in schools.

WatchGuard SchoolMate is an affordable system that meets all four security challenges to support productive classroom use of the Internet. It protects students and educators from falling victim to Internet abusers of all kinds, as it plugs security holes as soon as it's plugged into the network.

WatchGuard SchoolMate's main components are these:

- The WatchGuard Firebox houses core firewall functions in a standalone device and plugs into a school network in

minutes. In contrast, software-based firewalls generally require two or more days for installation and can carry a five-figure price tag. In addition, WatchGuard can serve any network-PC, Macintosh, or a cross-platform environment.

- WebBlocker software, which relies on Microsystems' CyberPatrol service, is highly regarded by K-12 educators as the most discriminating "guidance system" for student Internet use. WebBlocker allows educators to establish times of restricted and unrestricted use and categories of sites blocked. The site-blocking feature also allows educators to customize these categories.
- The WatchGuard Graphical Monitor module shows real-time graphical representations of host-to-host activity on the school network, enabling educators to see which sites students visit and what they do there. It plots connections so educators can monitor the composition of their network traffic. The Graphical Monitor module also measures the bandwidth being used by the school network and provides instant replay of network activity.
- The WatchGuard Historical Reports module keeps track of student's Internet activities by providing daily, weekly or monthly reports in an easy-to-read summary format. It produces "suspicious activities summaries" that serve as an early warning system of potential security breaches.

### Tip:

For more detail on the challenges of Internet use in schools and WatchGuard SchoolMate's role in overcoming them, check the paper entitled, "Surfing Schools: Issues and Answers regarding Students on the Internet" at <http://www.watchguard.com/schoolmate>.

## WatchGuard's VPN Wizard

Virtual Private Networks (VPN) is a standard feature of the WatchGuard system. The combination of VPN-enabling software-User Authentication combined with Remote-User VPN-in WatchGuard's standard bundle of security features makes it the first company to provide protection for an extended network to remote users at no additional cost. WatchGuard also offers Branch Office VPN software for companies whose network includes multiple locations, such as branch offices.

Activating the Remote User VPN to include mobile workers merely involves clicking on a dialog box in the WatchGuard Security System software. The Remote-User VPN component of WatchGuard's standard system relies on Microsoft's industry standard Point-to-Point Tunneling Protocol (PPTP).

Windows NT 4.0 and Windows 95 machines are either equipped with PPTP or are PPTP-ready (can run Dialup Networking 1.2), so users of the WatchGuard system can have literally no additional costs if they wish to extend their secure network to include mobile workers. To use the VPN, workers on the road dial into their ISP or corporate network via standard remote access. A "tunnel" is created with the PPTP. All traffic then flows transparently through the secure tunnel across the public network.

The complexity of setting up a virtual network of branch offices is simplified too, with the VPN Wizard. Like the 'wizards' that accompany much Windows software, the VPN Wizard guides you through a set-up process. In this case, it simplifies establishing the VPN no matter how many branch offices are included in the extended network.

The VPN Wizard enables you to establish the Branch Office VPN with point-and-click ease, as the Wizard steps through the process of setting up remote sites and configuring the remote Fireboxes for VPN - all from a single location.

## Systems Requirements:

The following is an itemized list of the minimum requirements recommended by WatchGuard Technologies Inc. to run WatchGuard:

- Pentium-class processor
- Minimum 16 MB Ram
- Windows 95, Windows NT or

- Linux network client
- CD-ROM
- 3.5" floppy drive
- Hard disk with 5 MB of free space (50 MB if same workstation is used for logging)
- SVGA display adapter and monitor
- Modem for pager notification (optional)

## AltaVista Software's Firewall 97 - The Active Firewall

AltaVista is dedicated to develop and market software products for use in the emerging, integrated Internet/Intranet business environment. Their portfolio of innovative software products enables you to:

- find useful information
- control access to information and transmit it securely
- collaborate and communicate from multiple locations.

AltaVista products and services are designed to integrate all levels of your working environment, from the Internet and enterprise, to workgroup and individual use, to allow location and platform-independent computing.

To increase global awareness of the AltaVista brand and showcase AltaVista software technologies and products, the company provides the already well-known AltaVista Search Public Service, which is the world's most popular Internet search engine, and other Internet services free on the World Wide Web. They also license their Internet services to major telecommunications and media companies outside the United States, and to major Internet content providers.

Figure 14.71 shows a screenshot AltaVista Firewall Center Web site.

### Note:

For more information, contact AltaVista Software Inc., 30 Porter Road, Littleton, MA, Tel.: 508-486-2308, Fax (508) 486-2017. Or you can visit their Web site at URL <http://www.altavista.software.digital.com>

## *AltaVista Firewall: Always in Motion*

The AltaVista Firewall keeps constant watch on the network day and night, actively deploying evasive action technology to detect and stop network attacks.

The active firewall offers maximum security based on a unique four-tiered alarming system. This alarming mechanism automatically takes actions not only on the attack itself but also on its context.

As a result, AltaVista Firewall provides better tools to fight against repetitive or multi-proxy threats. Furthermore, AltaVista Firewall 97 also provides a wide spectrum of actions to respond to any attack levels. This includes mail or paging to system administrators, custom scripts, and even services or firewall shutdown to guarantee the protection of your assets under any circumstances.

According to AltaVista Software, their firewall is quick and nimble enough to be called the Active Firewall. It's the only one that independently reacts to network violations while alerting you via pager, e-mail, or audio alarm, and even shutting down the firewall against heavy attacks.

AltaVista Firewall 97 for Windows NT provides a flexible and secure connection between your private network and the Internet, or other insecure public TCP/IP networks. It prevents unauthorized access to your private network, while providing controlled access to Internet services to users within your network. According to Data Communications Magazine (March 21, 1997) AltaVista Firewall 97 "shines in ease of management."

According to the vendor, this firewall is the only one that takes an active role in your security management. With its unique intelligence, it warns you of impending danger of intrusions, is constantly looking for threats to your defined security zone, and takes evasive action when attacks do occur. Figure 14.71a is a screenshot of Firewall 97's main menu.

AltaVista Firewall 97 combines trusted application gateways, comprehensive logging, reporting, real-time alarms, strong authentication, graphical user interface (GUI), and a step-by-step installation wizard all in one software package. Also, according to my lab tests AltaVista is by far the fastest firewall available in its class, with no compromise on security. This demonstrates not only its high efficiency, but the tightness of its Windows NT integration.

## ***Services: a Matter of Security***

AltaVista' firewall provides trusted application gateways to allow users access to most common services on the Internet, including file transfer (FTP), remote sessions (Telnet), World Wide Web, Mail, News, SQL\*Net, RealAudio and finger. This firewall can also be configured to allow controlled access from the internal network to the public network, and also from the public network into the internal network. Figure 14.71b shows a screenshot of the alarms for e-mails featured by AltaVista.

It also enables you to customize generic TCP application gateway, which provides secure connections to services that do not use a dedicated application gateway.

## ***Security: Supporting SSL***

AltaVista's firewall also supports the Secure Sockets Layer (SSL), which is included with the World Wide Web proxy.

Its security model is enforced at several levels. IP forwarding is disabled and continuously monitored by the firewall alarm system. All access through the firewall must be through the trusted application gateways. A system on one side of the firewall cannot access another system on the same side of the firewall via the firewall.

This firewall also has strong authentication support, using one-time passwords. FTP and Telnet gateways can be configured to allow access only to authenticated users through NT domain login or with hand held authenticators. The hardware authentication cards such as SecureID cards from Security Dynamics must be purchased separately.

There is a comprehensive logging of all events relating to the operation of the firewall that is worth mentioning. The reports it generates, which gives a summary of the usage of the firewall and of individual services are excellent. The reports can be viewed through the user interface, mailed automatically to a specified distribution list at regular intervals, or both. All reports use information from the system log files. A wide range of summary reports and detailed reports are also available.

## ***Management Features: Remote Management Through Tunneling***

AltaVista's firewall has an active architecture which can take actions on behalf of the system administrator with a sophisticated alarming and notification system.

It has automatic alarms that alert the system administrator to unusual or potentially threatening events relating to the firewall. The alarm system continually monitors the firewall system in real time for any events that are unusual or suspicious. Standard alarm actions, which includes sending mail to the system administrator, raising the security status of the firewall, triggering a custom script and shutting down individual services or the whole firewall are also one of the main features of this firewall.

Because system administrators may have to manage several platforms, the remote firewall management is very consistent and compatible on all supported platforms. It implements a HTML based user interface for a same look-and-feel. It is written in Java for enhanced portability.

AltaVista Firewall 97 offers remote management for firewalls within any network sizes from a centralized console running either Windows 95 or Windows NT. This is both a cost and time saving feature which allows system administrators to monitor and take quick actions on their UNIX or NT based firewall.

Remote management is also offered, which allows system administrators to perform the following operations remotely:

- View/Change firewall status
- View firewall activity
- View firewall event messages
- Stop/Start firewall services

When thinking about remote management of firewalls, you must be careful with the side effect of it: the establishment of a weak link to the firewall via a serial port or Telnet session on a high port. With AltaVista Firewall, its remote management services is done through tunneling, using AltaVista's Tunnel. The tunnel product provides RSA 512 bit authentication, MD5 integrity and the strongest encryption worldwide with RSA 128bit (U.S.) and 56/40 bit (International.)

The new remote management enables system administrators to view firewall activities and allows them to quickly take appropriate actions. Consistently with the OnSite Computing vision of AltaVista, you are able to manage the firewall from anywhere within the Intranet or from an untrusted network.

On all supported platforms, the remote management displays the states of all services as well as various statuses and alarms. It also allows to modify the firewall status and start/stop specific services such as FTP. Additionally, on Digital UNIX, network administrators can maintain and manage security policies, user authentication, DNS, mail, SNMP alarms and active monitoring of traffic. Furthermore, different levels of control can be assigned on UNIX. As an example, one Firewall administrator can monitor the status of the firewall, while another can change some security policies.

The installation wizard provides an easy step-by-step firewall installation, including DNS configuration. Its comprehensive graphical user interface through which all configuration administration, and management tasks are performed makes management of the firewall much easier.

Another great feature is its automatic shutdown of individual services, or the whole firewall, if the firewall is under continued or repeated attack. AltaVista Firewall for Windows NT can automatically shut down the service or the whole firewall to prevent the firewall from being compromised.

## ***URL and Java Blocking***

This is both a performance and a security feature. According to easily definable policies, AltaVista Firewall 97 can block URLs to preserve network performance and to restrict access to specific Web sites for productivity purposes. Security managers can define specific policies for URL access. AltaVista Firewall 97 can also detect and block Java applets entirely by allowing selective filtering of Java applets through the firewall to protect against one the most common network attacks.

## ***Enhanced Proxy***

The firewall has an updated proxy contains significant performance improvements based on code optimization and caching implementation. It supports the following protocols:

- HTTP,
- HTTPS/SSL,
- gopher and
- ftp.

It implements the CERN/NCSA Common Log Format for enhanced reporting and integration with third party analysis tools. As for other proxies, access restriction policies per user can also be combined with time limitations.

Support for Real-Audio proxy: RealAudio is an application that allows playback of audio in real-time over Internet connections. Through the RealAudio proxy, managers can allow or prevent users on internal network systems with Web browsers to access RealAudio services on the external network. For this proxy, system administrators can specify security policy details, time restrictions and blacklists of hosts forbidden access (common with ftp, Telnet and finger proxies.)

A new generic UDP proxy allows UDP-based applications, such as Internet Chat, to pass through the firewall securely. Also,

with AltaVista Firewall 97, you are a system architect, you are now free to build any sophisticated, distributed networks of Oracle7 or third-party data repositories across the Internet. SQL\*Net establishes a connection to a database when a client or another database server process requests a database session. The proxy is based on the Oracle Multi-Protocol Interchange (MPI), so it inherits many of the Multi-Protocol interchange's features.

SQL\*Net firewall proxy is able to control access based on information contained in the SQL\*Net connection packet. This includes the client machine name, the destination name and the database service. The firewall also integrates the administration of this authorization list with various authentication methods such as smartcards.

AltaVista Firewall 97 broadens security policies by offering a generic TCP relay for one-to-many and many-to-one connections. Consequently, an instance of the generic relay such as news can have one server on the inside of the firewall getting feeds from multiple news servers on the outside.

This generic relay is also fully transparent outbound so there will be no need to reconfigure internal systems. The management GUI supports both one-to-many and many-to-one configurations.

## ***Powerful and Flexible Authentication***

The enhanced WWW proxy includes authentication for specific users or group of users by any authentication schemes currently supported by the UNIX firewall such as CRYPTOcard or re-useable passwords. This feature provides system administrators with great flexibility to implement their policies with finer granularity. This authentication is integrated with the existing system management GUI on UNIX.

AltaVista also integrates Windows NT domain authentication scheme onto its firewall. This allows access to Internet services (e.g. FTP, Telnet) to users authenticated by this scheme and finer grained control over firewall traversal. This is a clear win for both end-users and MIS managers. MIS managers can easily integrate NT domain concept in their policies and users can appreciate a simplified login mechanism. The AltaVista Firewall 97 authenticates in both directions across the firewall.

## ***Dual-DNS Server***

Before the introduction of AltaVista Firewall 97, the recommended name server configuration was the hidden DNS setup hiding the internal address space from the untrusted network. However, this recommendation required to set up a second name server within the Intranet causing some management issues.

With AltaVista Firewall 97, firewalls can now be configured as Dual-DNS servers that understand which name services are internal or external. This Dual-DNS server is fully configurable through the GUI based management.

Most of us, Internet Managers, are mostly interested in dedicated boxes for security, performance and management reasons, correct? Well, AltaVista has been offering the capability of running a security low-end server on the same UNIX box. It managed to minimize any security impacts by a close integration between those two products. With Firewall 97, AltaVista now extends this integrated solution to Windows NT servers.

### **Note:**

Note that the Windows NT server must be connected to the ISP through a router. Support for a direct connection over an ISDN or a dial-up line is not yet available in this firewall but according to the vendor, will follow in a next release.

## ***DMZ Support***

With DMZ (Demilitarized Zone), AltaVista 97 on UNIX offers more than a simple trusted/untrusted implementation supporting only two LAN connections. While two interfaces is often enough for an Internet-oriented firewall, many organizations need three:

- one for the Internet,
- one for public servers for such items as WWW, News and File Transfer Protocol (FTP), and
- one for the Intranet.

The introduction of DMZ support provides security managers with great flexibility when configuring their security implementations. While DMZ is fully supported, it still needs to be done outside the GUI. An application note in the GUI describes the configuration process.

**Tip:**

AltaVista Firewall can be expanded to handle larger, more complex environments as it supports a large variety of platforms, including Windows NT, BSD/OS and Digital UNIX, which enable it to easily scale from small businesses to enterprise environments.

## Configuration

AltaVista Firewall software can be used with the AltaVista Tunnel product to create a virtual private network over the Internet and allow encryption and authentication securely through the AltaVista Firewall. Both products can run securely on the same system with a packet filter application provided with the firewall.

**Note:**

For more information on AltaVista's Tunnel product, check the URL <http://www.altavista.software.digital.com/tunnel/index.htm>.

The product supports Remote Access Service (RAS) on NT for external connection. This feature is used most often in an environment where Internet connection is via a dial-up line.

## Hardware Requirements

The AltaVista Security Pack 97 contains all firewall proxies, firewall remote management, and full authentication, with no extra costs. It consists of a complete AltaVista Firewall 97 kit and a complete AltaVista Tunnel 97 kit.

The systems requirements are:

- **System:** Pentium
- **Disk space required for installation:** 40MB
- **Disk space required for use:** 2GB
- **Memory RAM:** 48MB, 64MB recommended for optimum performance.
- **OS:** Windows NT V4.0 Service Pack 2 or later required.
- **Browsers:** Netscape Navigator 3.0 or Internet Explorer 3.0
- **NICs:** 2 interface cards with static IP addresses.

**Note:**

SQL\*Net proxy does not run on Alpha platforms running Windows NT



# ANS Communications's InterLock Firewall - a Dual-Homed Application Level Gateway

Advanced Network & Services, Inc., the former parent company of ANS CO+RE Systems, Inc. (ANS), was established as a not-for-profit company in 1990 by IBM, MCI and Merit -- a consortium of Michigan universities. Its mission was to advance high-speed networking technology and use. In 1994, Northern Telecom also became a member. As the principal architect of the National Science Foundation Backbone (NSFB) network service, ANS developed proprietary expertise in the design, development and deployment of large-scale, high-performance, wide area data networks.

Company founders recognized that the acceptance and adoption of this new technology by the business community would be critical to the overall success of the Internet. They established ANS CO+RE Systems, Inc., in June 1991 to target the networking and security needs of the business community. ANS' nationwide backbone enabled large segments of the Internet to carry commercial traffic. During the following four years, use of the Internet by commercial organizations skyrocketed.

In 1995, America Online (AOL) acquired the assets of ANS CO+RE Systems, Inc. As the nation's fastest growing provider of online services, AOL was impressed with the success ANS had in deploying and operating large scale, private networks and sought to use ANS's networking resources to better serve their rapidly-growing customer base.

ANS uses its expertise to deliver high-speed, value-added internetworking solutions that meet the mission-critical requirements of businesses and other organizations.

ANS offers services in three areas: Enterprise Networking Services, Web Application Hosting & E-Commerce Solutions. ANS designs, engineers, installs, manages, monitors and maintains nationwide private corporate data networks over one of the fastest, largest TCP/IP networks in the world. It is dedicated to helping businesses achieve their full potential through custom-designed internetworking solutions and through the use of resources available on the Internet. ANS is also committed to focusing on network security and offering unparalleled support services to its customers.

Since its formation, ANS has been a pioneer in the Internet and has led the industry in implementing higher performance networks and the scaling of large IP networks. ANS people designed, deployed and managed the construction of -- the first full duplex public 45 Mbps data network and a major backbone network of the Internet. The ANS team was the driving force behind several advanced routing technologies which enhance the scalability (i.e., the ability of the network to work efficiently as the number of users and the amount of traffic increases dramatically), and thus the overall reach and performance of the Internet. ANS also supports the largest closed user group in the world - 8 million America Online subscribers. Figure 14.72 is a screenshot of ANS Web site.

**Note:**

For more information, contact ANS at 1875 Campus Common Drive, Suite 220, Reston, VA 20191-1552. You can contact them by phone at 800-456-8267 from within the US or +1-703-758-7700, Fax: +1-703-758-7717. You can also send an e-mail to [info@ans.net](mailto:info@ans.net) or visit their Web site at URL <http://www.ans.net/>

## ANS InterLock

ANS InterLock Firewall Service provides network access control, attempted intrusion detection/response and cost accounting functionality to help organizations protect and manage valuable Intranet and Internet resources. One of the original application-layer firewalls, it provides high granularity of control with a full line of application proxies for all the major TCP/IP services as well as address remapping, file integrity monitoring and a real time utility to detect and prevent intrusion attempts. Detailed auditing information, cost of use/abuse controls and accounting reports are provided for advanced management of network resources.

As discussed throughout this book, firewalls are an important component of any organization's network security architecture. Good firewalls provide security controls without making Internet access prohibitively difficult for the end user. Better

firewalls improve upon those solutions by adding detailed audit trails and accounting information. State-of-the-art firewalls offer management control over secure Internet and Intranet resources. In short, they combine access control mechanisms, detailed logging, usage and chargeback reports, intrusion detection capabilities and graphical administrative interfaces to provide secure, managed access network solutions. ANS InterLock service 4.0 has evolved to meet customer requirements for this advanced level of security, accountability and manageability. Figure 14.73 shows a layout of a multi-ANS InterLock configuration

## ***ANS InterLock Service***

The ANS InterLock service is a connectivity management tool that provides access control, intrusion detection and cost accounting functionality. Configured as a dual-homed application level gateway, the ANS InterLock service manages access between site-designated protected and unprotected networks. Proxy support is provided for an expanding list of Internet applications including:

- FTP
- News (NNTP)
- TN3270
- Gopher
- Real Audio
- X-Windows
- HTTP (Web)
- SMTP
- Generic TCP
- LPR/LPD
- SSL
- Generic UDP
- Network Time Protocol
- Telnet

ANS InterLock solutions can be deployed throughout an organization. Figure 14.73 above shows a multi ANS InterLock configuration for the XYZ Corporation. XYZ uses ANS InterLock systems to manage Internet connectivity, to isolate R&D information from unauthorized corporate users and to limit access to internal resources from Intranet-connected vendors.

As a network security and resource management tool, ANS InterLock service provides:

- Application Gateway Services Between IP Networks;
- Access Controls by User, Group, Pair of Hosts or Networks,
- Protocol, Time of Day;
- Cost of Use (Abuse) Controls and Reports;
- Attempted Intrusions.

## ***Enhanced features in Version 4.0***

ANS InterLock service 4.0 offers enhanced security, performance and functionality over previous releases. The following is a list of advancements offered with this release:

- **High Performance WWW Gateway** - Improved performance, password changing from Web browser, Java-filtering and URL-level controls. Performance tuning and customization are possible with this gateway.
- **HTML Reports** - Audit tools which generate HTML and ASCII based output for an expanded set of user, group, chargeback and protocol reports.

- **Attempted Intrusions** - Real time log monitoring system to watch the logs for a variety of attacks including port scans, IP spoofing and ISS or SATAN probes.
- **Integrity Watcher** - Utility to monitor the permissions and contents of key files on the system from potential tampering.
- **SSL Forwarder** - Support for SSL based forwarding (https: and snews: URLs).
- **RealAudio 2.0 support** - Support for Real Audio's RA Player.
- **Solaris 2.5 port** - Overall system performance and stability improvements and support for UltraSparc platforms.
- **HTML-based Administrator Interface** - Security policy updates via Web-browser.
- **Enigma Logic Support** - Support for Enigma Logic DES Gold card authentication.

## InterLock's Access Controls

A primary function of any firewall solution is allowing access to users with appropriate privileges while preventing unauthorized transactions. Traditional firewall solutions rely on IP address and protocol as the sole criteria for deciding if connection requests should be granted or denied. The ANS InterLock service provides access controls on user and system administrator activity at a highly granular level.

The following is a description of the overall security model, address hiding, administrator controls and the access control rulebase of ANS InterLock:

- **Security Model** - The ANS InterLock system integrates both a modified (not just hardened) operating system and the set of associated applications. ANS has obtained and modified the SunSoft Solaris operating system source code to improve security and overall system performance. The general security model is: that which is not expressly permitted is denied. This model is implemented using application proxies which grant/deny access requests based on queries into a central access control rulebase (ACRB). By default, ANS InterLock application gateways require user level authentication.
- **Information Hiding** - The ANS InterLock service supports the use of RFC1597 and other non-NIC assigned IP addresses on the protected network. Even though, the remote user will perceive that he/she has an end-to-end connection to a remote host, the application proxy is managing two connections; one from the client application to the ANS InterLock and a second from the ANS InterLock to remote server. Under this model, the original source address is hidden from the remote destination and vice versa. All connections will appear to be coming from the ANS InterLock network interface nearest the destination host. The system also can be used to remove or remap domain name information from outbound mail and news articles. This further controls leaks of potentially useful information to an attacker about the architecture of the protected network.
- **Separation of Administrator Privilege** - ANS InterLock service 4.0 supports multiple administrator privileges which can be assigned to different accounts. One or more of the following administrator privileges can be assigned to an account.
  - **Mail** - configuration/control over mail system
  - **Security** - maintenance of security policies (rulebase)
  - **Admin** - creation/maintenance of user accounts
  - **Audit** - monitoring/data reduction of log information
  - **System** - miscellaneous other privileged system maintenance operations
- **Access Control Rule Base (ACRB)** - Each application gateway makes queries into the ACRB to determine if a connection request should be granted and, if so, the level of service which should be provided. ANS InterLock administrators define the set of rules which describe an organization's security policy. There are multiple components to each rule. The first portion of each rule describes the situations when the rule is to be enforced. Rules which do not match a particular situation (e.g. outside the time range) can be configured by the administrator to deny access or simply remain inactive. The second part of each rule defines the authorizations or constraints to be enforced. Different levels of logging (Low, Medium, High, Debug, Trace) can be associated with each rule.
  - **Access Controls Criteria**
    - User or Group

- Protocol/Port Number
- Source and Destination Address Associated with Connection
- Time of Day (start/stop times)
- Days of the Week
- **Rule Constraints**
  - Direction of Connection/Data Flow
  - Authentication Required (SecurID, Enigma Logic, Unix password)
  - Audit Level (Low, Medium, High, Debug)

When making changes to the ACRB, the name of the administrator making the change and a timestamp are associated with each rule. This feature is useful for multiple administrator coordination and accountability. Figure 14.74 shows a sample rulebase modify screen of InterLock.

- **Application Gateways** - One of the original design goals of the ANS InterLock service was to develop application proxies which would require user authentication. This was easy for some gateways (e.g. FTP, Telnet) since user/password mechanisms were included in the protocol specification. For applications like SMTP or NetNews (NNTP), the ANS InterLock system uses a concept of mapping entries to have user-level controls even though those services are normally non-authenticated. For access to applications via Web browser, the ANS InterLock system takes advantage of proxy and basic authentication mechanisms to require passwords for these transactions. There were several reasons for this approach, more granular control, more detailed auditing and chargeback reports based on user, group and/or IP address. Below a typical Web transaction is traced.

Web access is transparent to the end user. The only requirement is to make the browser aware of the ANS InterLock through standard proxy configuration as shown in figure 14.75. What is unique about this approach is that the Web gateway on the ANS InterLock prompts the user for name and password whenever a remote access is requested via the desktop Web browser, as seen on figure 14.76). Most browsers cache this information for future requests. Even though each Web transaction is separately authenticated, the user enters his/her password only one time.

## ***InterLock's Access Management***

InterLock's access management describes the audit, control and reporting functions such as audit levels, limitation of access to non-business related sites and so on.

### **Audit Levels**

It is common for sites to require more detailed information on some transactions but less for others. Audit levels can be assigned to each rule added to the ACRB. For example, medium auditing may be required for corporate users accessing the Internet but a much higher audit level may be assigned for vendors accessing internal resources.

### **URL-Level Controls**

Recognizing that site administrators are often concerned about the percentage of traffic going to non-business related sites, the ANS InterLock service provides support for restricting users from going to specific URLs in the WWW gateway. Since many Web sites today are implemented using multiple hosts with different IP addresses, this blocked site database allows URL-level controls for pages, directories or entire sites without having to add an excessive number of rules into the ACRB.

### **Log Files**

ANS InterLock 4.0 includes a modified version of the Unix syslog daemon. Each service generates logging information allowing an administrator to generate usage statistics, isolate configuration problems, and determine if there have been any attempts to obtain unauthorized access to the protected network.

Log entries contain information specific to each service including the time the action occurred, a unique process ID associated with the connection, number of bytes sent in each direction, the type of message, the addresses of the source and

destination host, the user accessing the service, any commands entered, and an informative message describing the action performed.

FTP logs include information on the operation performed (put versus get) and the name and size of the file being transferred. HTTP entries contain information on URL accesses and byte transfer sizes. All user and administrative activity is logged. Audit information can be logged to local disk and to a syslogd on a protected site host. Figure 14.77 shows a typical HTTP log entry.

## InterLock's Reports Feature

A number of reports and formatting options are available with ANS InterLock. Reports can be configured to generate HTML or ASCII output, HTML-based reports can be viewed by system administrators via local Web browsers. The system generates usage reports by user, group, IP address, and protocol. Web usage is tracked via additional reports which identify top surfers, top sites accessed, and a list of users accessing non-business related sites.

## ANS InterLock Service For Intrusion Detection

The ANS InterLock service provides intrusion detection mechanisms to automatically notice and respond to potential attacks. The following is a list of the main InterLock features on intrusion detection:

- **Audit Log Thresholder** - ANS InterLock service includes an intrusion detection facility. The audit log thresholder is designed to look for administrator defined patterns in the system logs and to trigger an automated response when that pattern event occurs. For example, three failed logins from the same IP address may result in a rule being added to deny access from that host, an e-mail page being sent to the ANS InterLock administrator and an SNMP trap sent to the site's network monitoring station.
- **IP Spoof Guard** - The ANS InterLock system maintains a routing table for address-to-interface comparisons. Protected side subnets and networks are defined in this table. If the ANS InterLock system receives packets from (what it believes to be) a protected side address on the public interface, the spoof guard is triggered, the event is logged and the packet is discarded. The spoof guard also monitors for public packets on the private interface.
- **Port Scan Detection** - The operating system kernel used by the ANS InterLock system prohibits IP forwarding, ICMP redirects and all forms of source routing through the box. These security controls prevent the ANS InterLock ACRB from being by-passed with IP packets. All connection requests must be handled by a proxy gateway. If no proxy is configured for a particular port, connection requests to that port are logged and denied. The ANS InterLock includes a port scan detection system as part of the Audit Log Thresholder package to identify when sites are being probed by Satan, ISS or other port scanning utilities.

## Summary of InterLock's Security Feature

The following are the main security features found on InterLock firewall:

- **Granular Control** - Access Control Rule Base (ACRB) enables administrators to define the set of rules that describe an organization's security policy. With ANS InterLock Service, that which is not expressly permitted, is denied. Access and authorization functions let administrators control the use of each application protocol according to various criterias, as well as support to "least privilege" by separating the administrator's functions.
- **Modified Kernel (not just hardened)** - The underlying source code has been modified to remove IP forwarding, ICMP redirects, and source routing functions. The ANS InterLock firewall includes a port scan detection system to identify probes by SATAN, ISS, etc.
- **Address Remapping** - The ANS InterLock firewall hides internal network addressing and topology information from the external network and allows use of non-NIC registered addresses on the protected network.
- **Java Filtering** - Enables network administrators to filter out JAVA use from a central point.
- **Spoof Guard** - Prevents hackers from exploiting protected site network addresses to gain entry.
- **Audit Log Thresholder** - Recognizes and responds to potential security attacks in real-time. Attack patterns can be pre-loaded by ANS or created by you. Sophisticated response options include e-mail, paging, SNMP traps, scripts and

customer programs.

- **Integrity Watcher Daemon** - Monitors configurable set of ANS InterLock files not ordinarily subject to change. This helps protect your network against Trojan horse attacks.

# Global Technology's Gnat Box Firewall - a firewall in a floppy disk

Global Technology Associates, Inc. (GTA) is a privately owned, U.S. corporation involved in the development of computer network security systems. Founded in 1992 as a small flexible company of highly motivated software engineers, GTA has evolved into a leading innovator of network security firewall products. The company's GFX Internet Firewall System was one of the first firewall systems to be certified by the National Computer Security Association and has been widely recognized as a rock solid security solution. With the introduction of the GNAT Box Firewall software, GTA has sought to meet the growing demand for a truly affordable network security system.

Global Technology believes so much in GNAT Box that they decided to host their web site through a GNAT Box system. Thus, if you were to access their site (see a screenshot of it on figure 14.79), you would find that their Web server resides on a Private Service Network (PSN), attached to a 3rd network card in a GNAT Box system. This server has an IP address of 192.168.5.2, but you can't see that address as it is hidden and translated to the GNAT Box's external IP address of 204.96.116.177. A GNAT Box facility called a "tunnel" is mapping all the Web access requests through the GNAT Box to their Web server. A network diagram of their GNAT Box configuration is provided on that site, as figure 14.80 illustrates.

## Note:

For more information, contact Global Technology Associates, Inc. , 3504 Lake Lynda Drive, Suite 160, Orlando, FL 32817. You can call 1.800.775.4GTA, or internationally, +1.407.380.0220, Fax: +1.407.380.6080. You can also contact them via e-mail at [gb-sales@gta.com](mailto:gb-sales@gta.com) or via their Web site at the URL <http://www.gnatbox.com/index.html>

## Getting to Know GNAT Box Firewall

You shouldn't have to pay for security features you will not use or do not want in a firewall product. GNAT Box was developed to provide a powerful, simple and affordable IP network security solution for organizations that would otherwise be forced to purchase an expensive solution or due without IP security altogether.

So lets start by outlining what the GNAT Box is NOT:

- A general purpose computer system, so:
- you can't log on to it (there is no user shell)
- you can't Telnet to it
- you can't use it for a mail server
- you can't use it for a web server
- you can't run any other software on it
- A Unix system, although it uses core technology from the Unix operating system.

At the heart of GNAT Box is GTA's network address translation and stateful packet inspection engine. This facility was originally developed for GTA's premier turnkey dual wall firewall, the GFX Internet Firewall System. The stateful packet inspection facility monitors every IP packet passing through the GNAT Box to guarantee that:

- Network address translation is performed for all packets passing through the GNAT Box.
- Only valid response packets or packets passing through user defined tunnels reach hosts on the Protected or PNS networks from the External network.

This facility is tightly integrated into the GNAT Box's network layer to guarantee maximum data throughput.

## **Outbound Packets from the Protected Network**

When an IP packet arrives on the GNAT Box's protected network interface, the engine determines where the packet should be sent and performs the necessary modifications on the packet (i.e. network address translation) if required and then routes the packet to the correct network interface. Translation is performed if the destination host is on the External or Private Service networks.

If translation is performed, the IP packet's source address will be modified to be that of the network interface that is the route to the destination IP address of the packet. When a response packet returns to the GNAT Box, the packet is inspected to determine if the packet is in fact a response on an active transparency circuit.

If the packet is accepted it is then modified with the originating reply IP address and routed on to the Protected network.

## **Inbound Packets from the External Network**

In its default configuration the GNAT Box does not listen for any unsolicited inbound packets. It only responds to reply packets, (those packets which are returning in response to packets that originated from the Protected or Private Service networks). If you need to allow unsolicited connections to internal hosts, use the GNAT Box's Tunnel facility.

## **Outbound Packets from the Private Service Network**

The Private Service network works the same as the Protected network, except that the Private Service network can not reach the Protected network. If a host on the Private Service network attempts to reach a host on the Protected network, the connection will be refused. Additionally the following message will be generated to the system console (and if syslog is enable to the log server):

"Warning: Attempt by PSN to access protected network."

## **How Tunnels Work in GNAT Box**

When an IP packet arrives at the GNAT Box and it is not a response packet for an active connection, the packet is compared against user defined Tunnels. If the destination IP address and port match the entrance of a Tunnel, a new connection is created.

This new connection will automatically change the destination address and port of all packets arriving on this connection to be those given for the end of the tunnel. Additionally all response packets originating from the Tunnel's destination host will have the source address and port changed back to the Tunnel's beginning as the packets leave the GNAT Box.

## ***Standard Features***

The following is an overview of GNAT Box Firewall at glance:

- Secure Network Address Translation (NAT)
- Firewall protection utilizing proven firewall technology
- Transparent network access for TCP and UDP applications
- Transparent network access for non-standard applications like:
- RealAudio/RealVideo,
- StreamWorks,
- VDOLive, VXtreme, etc
- Simple to install and operate
- Supports more than 16,000 concurrent connections

- No limit on the number of users
- Web browser user interface
- Minimal hardware requirements
- Cost effective
- IP aliasing
- Dynamic and static address mapping
- Tunneling
- Protected Private Service Network (PSN)
- High performance
- PPP support
- Built-in support for several application protocols such as,
- RealAudio/RealVideo
- Xing StreamWorks
- VDOLive
- Vosaic
- CU-SeeMe
- VxTreme
- Vivoactive
- NTT AudioLink
- NTT SoftwareVision
- Real Time Streaming Protocol (RSTP)

**Note:**

How Do You Pronounce GNAT Box?

GNAT Box is pronounced, "nat box", with the 'g' silent, like the tiny insect called a gnat. The derivation comes from GTA's Network Address Translation.

## ***What is GNAT Box Firewall?***

The GNAT Box system is based upon GTA's GFX Internet Firewall Network Transparency technology, which has been distilled and refined to fit into a compact powerful software system. The GNAT Box system completely hides all IP numbers on an internal network from an external network, (typically the Internet). This feature allows organizations to use unregistered IP addresses or RFC 1918 addressees on the internal private network.

The GNAT Box is a firewall. The GNAT Box protects the internal network from unauthorized access, while allowing users on the internal network transparent access outbound. The GNAT Box in default operation offers no services to the external network, so there are no cracks to allow an intruder access to your internal network. The GNAT Box utilizes the GFX Network Transparency technology which maintains stateful information about all packets passing through the GNAT Box gateway and only allows returning packets that have been registered to pass back through to the internal network.

One of the great things about the GNAT Box system is its simplicity. The GNAT Box system is software that you run on your own hardware. No need to pay extra for hardware when you probably have the required hardware components already. The system boots and runs off a single 3.5" floppy diskette, you don't need a hard disk. All the hardware you need is:

- 386 CPU or better,
- 8Mb RAM,



- 3.5" 1.44Mb floppy drive, and
- 2 network cards(10 and 100mbp Ethernet and FDDI).
- Optional 3rd network card for a Private Service network.

**Note:**

## Ethernet Card Notes

Network interfaces are addressed by their two or three character device identifier and a positional number starting at zero. The first card of a specific type identified by the system will have a positional identifier of zero, (e.g. de0). If a second card of the same type as the first is found then it will have a positional identifier of one, (e.g. de1) and a third card will have a positional identifier of two, (e.g. de2). Each new type of card identified in the system will begin with a base identifier of zero. This naming scheme does not apply to cards that must be configured to specific values listed below.

The system doesn't require a keyboard or monitor for operation, however you'll need them for the initial configuration. Figure 14.81 shows a typical layout of a network using GNAT Box firewall.

**Note:**

## Considerations about ISA Cards when using GNAT Box

1. Network cards do not have to be of identical make and/or manufacturer.
2. Configure the network cards using the configuration programs supplied with the network cards. It is very important you configure the cards correctly or you may have problems later.
  - Plug and play should be off
  - Configure the interface type if you are using a combo card
  - Use the listed IRQ, PORT and memory address (if required)
1. ISA cards must be configured to operate in the 16 bit mode.

GNAT Box configuration is simple too, 4 commands (well 5 if you count the reboot command) to get the system up and running. Figures 14.89, 14.90, 14.91 and 14.92 shows a sequence of GNAT Box console configuration interface.

Once the system is up just use the web browser interface to administer the system, (if you need to). The GNAT Box configuration is simple yet powerful, facilities are provided for: static routes, IP aliasing, logging and inbound tunnels. According to the vendor, other features such as filtering will be offered in a later release.

For those organizations that need to allow some inbound connections, the GNAT Box offers a tunneling facility. This facility allows a service port (IP port) on the external network interface of the GNAT Box to be mapped to a port on the PSN network (with optional 3rd network card) or an internal host system. Facilities that you might want to tunneled include email, http (WWW), ftp and Telnet. Using the IP aliasing facility in conjunction with the tunneling facility the GNAT Box can operate in a virtual hosting role.

The GNAT Box system is cost effective. The hardware required is inexpensive, there are not many components that can fail, and there are no license restrictions on a per user basis as found on most other systems. Figure 14.82 gives you a graphic description of what you would need as hardware requirements to run GNAT Box firewall.

Figure 14.83 shows a basic GNAT Box firewall configuration, where the requirements are,

- Two Networks
- External Network (typically the Internet)

- Protected Network
- Operational Mode
- Unsolicited packets from the External Network are rejected.
- Packets that originate on the Protected Network are allowed to pass through the GNAT Box and their reply packets are allowed to pass back to the Protected Network.

Now, you can have a more advanced configuration (see a basic installation of GNAT Box on figure 14.85) for the GNAT Box. Figure 14.84 shows a typical example of such a configuration, where you have,

- Three Networks
- External Network (typically the Internet)
- Protected Network, and
- Private Service Network
- Operational Mode
- Unsolicited packets from the External Network are rejected.
- Packets that originate on the Protected Network are allowed to pass through the GNAT Box and their reply packets are allowed to pass back to the Protected Network.
- Tunnel(s) are defined to allow External Network access to servers on the Private Service network (see figure 14.86). Common servers might be web, email (see figure 14.87), and ftp.
- Users on the Protected Network have complete access to the Private Service network, as it is typical of a University or a multi-departmental company, as seen on figure 14.88.
- The Private Service network has no access to the Protected network unless a Tunnel is defined.

## Network-1 Software and Technology's Firewall/Plus - a High Performance Multi-Protocol Firewall

Network-1 was incorporated in July 1990. The company has an impressive credential, averaging 16 years of technical experience. In particular, Dr. Bill Hancock is a noted authority on networks, connectivity and security. He has published many books and is currently the network editor for Digital News & Review.

The company has designed, planned, audited & implemented over 3,000 networks worldwide. Their consultants have also conducted seminars at industry conferences (i.e. DECUS, TCA, INTEROP, CSI..etc) for many years and are all well-known speakers around the world.

Their experience spans many different hardware systems, including IBM, DEC, Sun, HP, PC's & Macintosh. Network-1 specializes in network and security software, consulting, training and seminars.

Figure 14.93 is a screenshot of Network-1 Software and Technology Inc. Web site.

### **Note:**

For more information, contact Network-1 Software and Technology Inc., at 909 3rd Ave. 9th Floor, New York, NY 10022. By phone at (212) 293-3068 or fax at (212) 293-3090. You can also contact them via e-mail at [sales@network-1.com](mailto:sales@network-1.com) or at their Web site at URL <http://www.network-1.com>.

# **About Firewall/Plus**

FireWall/Plus is a NCSA certified frame, packet and application filtering network security firewall. It provides a very high degree of security between internal corporate networks as well as controlling access to and from external networks such as the Internet.

Installation and configuration of FireWall/Plus is accomplished with a minimum amount of effort using a powerful Graphical User Interface (GUI). Using pre-defined rule bases the system can be installed in a plug-and-play manner and made available for immediate use. Since FireWall/Plus is transparent to the network community all network applications will operate without interruption or modification.

FireWall/Plus may be configured in a variety of methods to provide a secure firewall installation for networks. The most common configuration is as a dual-homed gateway, as shown on figure 14.94

In the configuration described on figure 14.94, FireWall/Plus provides total filtration services between an exterior network, such as the Internet, and the internal network. This is the first line of defense against unwanted network attacks.

However, for sites that require systems such as Web Servers and gopher servers to be accessed from internal users and external users, a demilitarized zone (DMZ) network configuration may be used, as shown on figure 14.95

This DMZ configuration would require two FireWall/Plus systems to secure the systems on the inside section of the network from both the external network and the DMZ systems.

## **Installation, Set-up and Use of FireWall/Plus**

Installation of the complete FireWall/Plus system (hardware and software) is accomplished with little effort. It involves the following basic steps:

- Selecting a default security policy rule base. FireWall/Plus provides a comprehensive set of pre-defined security policy rule sets from which to choose in order to dramatically reduce the amount of time it takes to set-up the firewall. Rule bases include e-mail outbound only, standard information services outbound only, file transfer outbound only, e-mail both directions, TELNET outbound only, web services access outbound only, gopher services outbound only, variations of the standard rule bases for specific site requirements and many others. This is accomplished via a drop-down menu item in the configuration section of the product.
- Obtain a license key from Network-1 technical support and insert into it into the product. This is accomplished via a dialog button on the main screen.
- Activate the product by clicking on the "Start Operations" tile from the main screen, as shown on the screenshot of figure 14.96.

## **Selecting a Default Rule Base for FireWall/Plus**

The default rule base of Firewall/Plus is very easy to be selected. By clicking on the Configuration tile or tab, the system brings up the Configuration Page, as shown on figure 14.97.

The Configuration File section at the top of the screen contains a drop-down menu with the currently loaded default security policy rule bases. You simply select one that matches the needs of the site and click the Save Settings tile at the bottom of the screen. The rule base is then loaded into memory and is ready for use.

The trusted side of the network is identified by a picture of an angel. The untrusted side is identified by a picture of a devil's head. This motif is carried throughout all Pages of the product to make it instantly obvious which side of the network a particular operation is affecting (an alternate set of icons are included in the package for those sites desiring a less dramatic identification of resources).

The trusted and untrusted network configurations both have a box called Block All Connections. This is the network "panic button" which is used in an emergency to immediately stop all traffic on either side of the FireWall/Plus(tm) product.

# Performance Statistics

FireWall/Plus provides real-time system and network performance statistics on system and network activities, as shown on figure 14.98. As filters and flags are added to the system and as the traffic loads increase over time, FireWall/Plus provides pro-active performance data so that the system may be upgraded before performance degradation occurs.

Additionally, network statistics for the trusted and untrusted sides of the firewall system provide detailed information on connections, node access counts and other items required for proper management of traffic performance.

## Additional and Advanced Filtering

As with any firewall, site customization will be required from time to time. FireWall/Plus allows a very high resolution filtering capability through the use of an intuitive and extensive GUI, as shown on figure 14.99.

Filters and flags may be added to any level of the protocol hierarchy, from the frame level to the applications level. By selecting a default configuration, a base set of filters have been set into the system. Using the GUI, the system or security manager may build custom filters and flags on top of the default configuration selected in order to implement specific site security objectives.

As an example, world-wide web (WWW) may be configured to allow outbound connections only via a browser such as MOSAIC or Netscape. By clicking on the filter status button, the symbol will change from the universal NO ACCESS (a red circle with a line through it) to a green check mark, as shown on figure 14.100, indicating that all users of the application on the trusted side of the network have the ability to make outbound connections to the untrusted side. If bi-directional connections for all users were to be allowed, the check mark would also be necessary on the untrusted side of the FireWall/Plus icon.

There are situations where advanced and detailed filtering and rules are required for specific network conditions or network resources. For instance, specific systems on the network, on the trusted or untrusted side, may require additional filters other than the general defaults. In this situation, very specific filters may be defined, as also shown on figure 14.100.

In the example of figure 14.100, a node named JOE with an IP address of 192.246.254.112 on the trusted network side is allowed only to use IP with a TCP transport with the 3com-tsmux application for inbound and outbound traffic. Further more, the FireWall/Plus system has been configured to only report on those packet conditions where the firewall did not pass the packets along.

For situations where the filtering described on figure 14.100 is not sufficient to solve an organization's security policy requirement, FireWall/Plus allows a to-the-bit-definition level filtering facility. Individual fields in the protocols may be identified and filtered based upon rule definitions and, if necessary, individual bits in a packet or frame may be toggled on or off to identify specific patterns of traffic to be filtered.

In the above example, field-level definitions are being set up for filtering. In some cases, bit masks will need to be identified in a frame, packet or application packet that will need a rule applied for filtering. FireWall/Plus, through the use of the GUI, provides a very simple manner in which to set up very sophisticated bit-level filtering (bit-level filtering provides application level functionality, such as proxy filters, without implementation of the application itself).

Figure 14.101 shows the set-up of a bit-level filtering mask for the destination service access point (DSAP) in an 802.3 (Ethernet) frame.

By clicking on the arrows above the bit-field, as seen on figure 14.101, the system or security manager may allow the field to be passed or rejected, depending upon the security policy required for the site. This type of granularity of filtering is usually very difficult to do with any firewall product and requires the writing of sophisticated scripts, usually in Perl. FireWall/Plus does this quickly and graphically without the hassle of learning a programming language.

# ***Summary of Features of FireWall/Plus***

As seen in this section, FireWall/Plus provides a comprehensive security solution with an easy to manage GUI interface. While incorporating the latest in security and expert technology, FireWall/Plus is a robust yet easy to use network security solution.

## **Technical Specifications**

Firewall Type:

- Frame, packet and application-level filtering
- Automatically blocks all traffic that is not allowed
- ("Nothing is permitted except that which is allowed")

## **Special Features and General Characteristics**

The following is a list of special features bundled with Firewall/Plus

- Very easy to set up (less than 30 minutes under most conditions)
- No special consultation or external services required
- Very low cost (includes hardware and software)
- Very high performance, real-time responsiveness
- Highly secure from external attack directly on the firewall itself
- Dynamic changes and updates means no downtime to users
- Ability to add additional protocols besides IP in the future
- Invisible to IP probes from external or internal networks
- Customer-configurable bit-level filtering capabilities

Firewall/Plus general characteristics:

Firewall/Plus is capable of defeating:

- TCP sequence number prediction
- Source routing
- RIP attacks
- Exterior Gateway Protocol
- ICMP attacks
- Authentication server attacks
- finger (firewall or internal nodes)
- PCMAIL
- DNS access
- FTP authentication attacks
- Anonymous FTP access (accept or reject)
- SNMP (to firewall or through firewall)
- Remote booting (firewall cannot be remote booted)
- IP spoofing
- MAC address spoofing
- Broadcast storms
- ARP spoofing
- TFTP to/from firewall and filter to/from networks

- Reserved port attacks
- Remote access
- External takeover from outside networks
- External compromise (firewall itself)
- TCP wrappers
- Gopher spoofing
- MIME spoofing
- Network analysis facilities
- Autoboot after power failure
- Autosave of set-up and parameters
- Autoboot into secure mode with or without manual intervention
- Prevents DNS manipulations
- Network traffic analysis
- Firewall performance statistics software
- Cross-charging facilities
- Undetectable intrusion trapping and reporting
- Security logging and analysis tools
- Non-detectable monitoring of firewall attacks

Firewall/Plus also provides logs of :

- all connections to/through firewall
- Extensive ad-hoc query facilities (make your own reports)
- External activities
- Accounting and chargeback reporting capabilities

The following are the management features and services provided by Firewall/Plus:

- Configuration files are in plain-text
- All "safe" outgoing connections are transparent
- Filtering and rule set-up is easy to implement
- Graphical User Interface (GUI) is very easy to use
- Little management or changes to firewall required
- Easy to use and maintain
- Modifications easy to implement
- Modifications to rules and filters are dynamic and immediate
- Handles large numbers of systems (thousands of nodes)
- Replacement code and updates take very little time (less than 1 hour)
- Robust hardware and software

These are the filtering capabilities of Firewall/Plus:

- Auto-disable UDP and SNMP
- Prevents source routing and IP forwarding through firewall
- Immune from RIP vulnerabilities
- Filters:
  - finger
  - ftp
  - gopher

- ICMP
- Mbone
- MIME
- NFS
- NIS
- NTP
- RPCs
- Redirect messages
- RIP
- routing protocols
- sendmail
- SMTP
- Telnet
- TFTP
- tunneling (assembly/disassembly)
- UDP
- WWW
- X11
- Xterm
- MAC addresses
- User-configurable application filters

## ***Systems Requirements***

To operate FireWall/Plus, you must have Windows NT Version 3.51 or 4.0, and NDIS 3.0 drivers for Ethernet/802.3.

The hardware requirements of FireWall/Plus are as follows:

- Intel Pentium or DEC Alpha class CPU, 133 MHz minimum clock speed
- 500MB disk space
- 1.44MB 3.5" floppy drive and/or CD-ROM drive
- 32MB of memory
- Video card, SVGA 14" monitor, keyboard, mouse
- NDIS 3.0 compliant Ethernet/802.3 Network Interface Card/s (SMC EtherPower PCI recommended)

## **Trusted Information Systems's Gauntlet Internet - an application proxy-based Firewall**

Trusted Information Systems, Inc. (TIS), has been dedicated to providing computer and communications security solutions for business information systems for over a decade.

TIS, a company with a worldwide presence, provides an unparalleled breadth and depth of security expertise. TIS products and services range from theory to practice, and policy to product, with a pragmatic approach. Through a combination of advanced research and engineering, system security analysis, practical and affordable solutions, and training, TIS is transforming the Internet into a safe place to do business.

TIS is the developer of the TIS Internet Firewall Toolkit. The firewall software allows system managers to control access between their corporate networks and internetworks. This firewall toolkit has been requested by over 50,000 Internet users.

TIS also offers a family of Gauntlet Internet Firewall products, and Gauntlet ForceField, a first-of-its-kind product designed to protect web servers. They have a patented RecoverKey technology that was developed to support effective, exportable, and recoverable software and hardware cryptography solutions. This technology allows software applications vendors the ability to provide strong data protection internationally and the end-user the ability to recover their data when their encryption key is lost, stolen, or destroyed.

TIS is internationally renowned for research in information systems security. They are actively participating in government research contracts and internal research and development projects that advance the state of the art in trusted system technology.

Under DARPA and National Laboratory sponsorship, TIS staff are performing innovative research in access control for O/S and networks, cryptography (including key management), security services for Internet mail, trusted distributed file systems, secure distributed operating systems, and integrated Fortezza support. TIS also provides trusted systems engineering and consulting to a number of major government organizations and DoD programs. Figure 14.102 is a screenshot of TIS Web site.

**Note:**

For more information, contact Trusted Information Systems, Inc., 15204 Omega Drive, Rockville, MD 20850. Or by phone at +1 (301)527-9500 or Gauntlet Sales at (888)FIREWALL (toll free) or +1 (301)527-9500, FAX: +1 (301)527-0482. You can also contact them via email at [tis@tis.com](mailto:tis@tis.com) or on the Web at URL <http://www.tis.com>.

## ***TIS Gauntlet Internet Firewalls***

Trusted Information Systems' (TIS) Gauntlet Internet Firewalls provide strong points of defense and controlled, audited access to services - both from within and without an organization's private network. Thousands of Gauntlet Firewalls are already in use internationally.

TIS' Gauntlet Family of Firewall products offers one of the most secure firewall system available today. The Gauntlet Firewall system is application proxy-based. By serving as the only connection between outside, untrusted networks or users and your private, trusted network, a Gauntlet Firewall uses specific software application gateways and strong user authentication to tightly control access and block attacks. Gauntlet Firewalls provide a network strong point where strict enforcement of your security policy is concentrated.

Since an application gateway is the most secure type of internetworking firewall, TIS has designed Gauntlet Firewalls to rely on proxies to provide services, as shown on figure 14.103. Therefore, no direct connection is ever made between machines on opposite sides of the firewall; network packets are never passed between the networks, only application data. Their unique design combines these seven tenets:

1. Simplicity in mechanisms and services provided
2. Simplicity in software design, development, and implementation
3. A "Crystal Box" approach, in which source code is distributed to allow for assurance reviews by customers, resellers, and other experts
4. No users are allowed on the firewall system itself
5. For a complete security audit trail, anything that can be logged, should be logged
6. Strong user authentication methods and mechanisms must be supported and encouraged
7. A firewall should enforce an organization's network security policy, not impose one of its own



# ***A Firewall Transparent to the User***

Gauntlet Internet Firewalls let you extend your organization's network security by establishing a Virtual Network Perimeter. Remote offices can network with your main office with the Gauntlet Net Extender. The Gauntlet PC Extender allows traveling users with remote access to become a part of your trusted network. And the Gauntlet Intranet Firewall allows controlled access between trusted workgroups inside your organization.

Gauntlet provide the transparency and ease-of-operation of filtering router firewalls, but the application-level security services strongly regulate both incoming and outgoing communications, as illustrated on figure 14.104. The proxy-based system of this firewall passes only application data, so security is assured. Gauntlet Internet Firewalls look like they are behaving as an internetwork router, but supply proxy-based security for specific provided services. Gauntlet updates include additional proxies as additional services are developed.

Gauntlet Internet Firewalls also allow multi-national companies to build Global Virtual Private Networks (GVPNs) over low cost Internet communications links. TIS Commercial Key Escrow (CKE) system allows corporations to utilize their own data recovery centers. Other services available on Gauntlet Internet Firewalls include Domain Name Service (DNS), a secure Web server, secure Anonymous FTP, and Internet electronic mail. Gauntlet Firewalls are IPSEC-ready, X.400/X.500 compatible, NSA MISSI approved, and DoD DMS compliant.

## **Note:**

What about GVPN?

Virtual Private Networks (VPNs) allow privacy for all allowed network traffic between two protected gateways through the Data Encryption Standard (DES). No level of trust between networks is assumed. But when a trusted relationship exists between networks, the security perimeters may be extended. Users can economically establish security-assured, high-speed, Internet VPNs at a fraction of the operating expense of dedicated, leased-line networks. Gauntlet Internet Firewalls come standard with software encryption; hardware encryption and Commercial Key Recovery are available.

As an add-on feature to Gauntlet Internet Firewall, a Gauntlet Intranet Firewall allows you to place additional network strongholds within your security perimeter, as shown on figure 14.105. You can pass authorized information quickly and securely inside your organization. It can be easily managed locally or remotely, using the same access rules and features provided by your Gauntlet Internet Firewall.

As far as firewall management, Gauntlet also includes:

- A secure, graphical management interface, accessible from an authorized computer on your trusted network.
- A firewall system integrity checker using cryptographic checksums to detect and report any changes in the system software.
- "Smoke alarms" that can be configured to "go off" any time connections to unsupported services are attempted.
- An audit tool that provides audit reduction and reporting on a timely basis.

## ***Extending Firewall Protection to Remote Offices***

Gauntlet enables you to extend your trusted network's security perimeter by using is key to the dynamic, flexible ways you work today. Using a Gauntlet Net Extender or PC Extender, all of the services and security of your existing Gauntlet Internet Firewall are extended to your remote offices and users through strong encryption.

## Gauntlet Net Extender

An add-on to your existing Gauntlet Internet Firewall, the Net Extender supports remote sites connected to a primary site by an untrusted network, using encryption to provide a private connection. The network security perimeter can be extended to allow remote access to all services. It is remotely managed and has the same features as a Gauntlet Internet Firewall. Figure 14.106 illustrates this configuration

## Gauntlet PC Extender

Also an add-on to your existing Gauntlet Internet Firewall, the PC Extender extends the network security perimeter from host-to-host or from hotel room to trusted network, allowing for privacy and easy access on business travel. Figure 14.107 illustrates how it works, through its interaction with the Gauntlet Internet Firewall employing the same strong cryptography for privacy, whether directly connected to the trusted (inside) network or dialed in. Strong authentication is required to establish trust when the user is outside the physical security perimeter.

## Technologic's Interceptor Firewall - an Intuitive Firewall

Technologic, Inc. is a leading provider of network security products and services for the Internet and Intranets. They are the developers of the Interceptor Firewall Appliance--a "plug and play" firewall including hardware and software--as well as other security products and services.

Figure 14.108 is a screenshot of Technologic's Web site.

### Note:

For more information, contact Technologic, Inc. 1000 Abernathy Road, Suite 1075, Atlanta, GA 30328. You can call 770/522-0222 or 800/615-9911, Fax: 770/522-0201. You can also contact them via e-mail at [info@tlogic.com](mailto:info@tlogic.com) or on the Web at URL <http://www.tlogic.com>

## An Overview of Technologic's Interceptor

Interceptor Firewall Appliance is an application proxy firewall designed to provide maximum network security in a turnkey package for companies with Intranets or Internet connectivity. Interceptor Appliance is a bundled solution including hardware and software that provides plug-and-play firewall security. Interceptor is a comprehensive firewall that protects from an organization's external Internet connection all the way down to the individual desktop. Interceptor is delivered with ready to use proxies for all leading Internet applications and services.

Interceptor 3.0, released in March 1997, includes many useful capabilities including,

- A Secure Wide Area Network (S/WAN) enabled version of Virtual Private Network (VPN),
- Compatibility with Microsoft's Proxy Server,
- Secure, enhanced Remote Administration,
- Diagnostics and Reporting (RADAR) management,
- Web-based management interface,
- Web caching,
- Management of multiple firewalls,
- Interoperability with other firewalls,
- Added security measures,
- Windows-based management reporting,
- Automatic paging and emailing for security alerts

- Ability to easily create a corporate Intranet within your existing network,
- 100% proxy transparency,
- User authentication for WWW access at the individual URL level,
- On-demand security scanning using Internet Scanner from ISS to verify security.

Interceptor Firewall Appliance is available for configurations supporting 32, 256, 1024, 4096, and unlimited network connections. It is delivered as a pre-configured hardware/software system. For organizations that have already designated a processor, a software-only version is also available. It is available in English, Chinese, and Japanese language versions.

Interceptor's reputation for being one of the most secure, reliable, and easy-to-use firewall on the market has made it a favorite among small and large organizations alike. Companies like Lockheed-Martin Corporation, BellSouth, GEAC (formerly Dun & Bradstreet Software), and Security First Technologies all use Interceptor to keep their information assets safe and accessible.

## ***Interceptor's Components***

The following is an overview of the main components and features of Technologic's Interceptor firewall

### **Virtual Private Networking**

As discussed on chapter 3 "Cryptography: Is it Enough?," strong and manageable encryption technology enables the use of the Internet for private network communications. We all are looking for cost-effective alternatives to expensive private networks and WANs based on leased lines, and Virtual Private Networks (VPNs) can be an alternative.

VPNs provide a protected private path for network traffic between two or more gateways. High-speed Internet VPNs can be established and maintained at a fraction of the cost of dedicated, leased-line networks.

Interceptor 3.0's fully integrated security solution for Intranets includes an S/WAN enabled version of VPN. S/WAN designates specifications for implementing the Internet Engineering Task Force's Internet Protocol security (IPSec) standards to ensure interoperability among firewall and TCP/IP products. With this interoperability in place, as shown on figure 14.109, users can securely exchange data with other companies or departments implementing other S/WAN enabled firewalls and systems.

### **Secure Encryption for All Applications**

Confidentiality is an important component of any network security policy. It is a vital issue for organizations leveraging the cost savings inherent in public networks such as the Internet. With Interceptor's VPN option, you can encrypt data from firewall to firewall and from client to firewall. Sending e-mail, transferring files, browsing a web site, or connecting to a remote computer can be performed in privacy using encryption over the Internet. And because Interceptor uses the IPSec standard for VPNs, you're assured of industry compatibility.

### **Transparent Encryption for Users**

Interceptor's VPN encryption is automatic and transparent to the individual user and does not require special or modified client application programs. The encryption takes place in the TCP/IP kernel at the IP level, which provides fundamental, lower-level security than higher-level protocols such as SSL and S/HTTP.

### **Internet Scanner**

A significant percentage of network vulnerabilities result from the presence of bugs, holes and system configuration weaknesses on devices attached to an organization's network. Technologic uses Internet Scanner from Internet Security Systems (ISS) - a powerful network scanning system - to locate these exposures. Internet Scanner identifies network security vulnerabilities on both internal and external machines.

Internet Scanner is the first and most comprehensive network security assessment tool available to help you close the gap

between security policy and security practice. Internet Scanner provides you with an excellent view of your network's security exposures. The system tests for over 130 known vulnerabilities and recommends appropriate corrective action. It also provides frequent updates with latest vulnerabilities and automatically identifies and reports these vulnerabilities.

## The Connection Manager

The Connection Manager is the first level of protection for Interceptor. It listens for connection requests for each service provided by Interceptor. Connections are accepted or rejected based upon the type of request, the source and destination IP addresses, and the time of day. Accepted connections are directed to service-specific gateway programs. Each connection request, whether it is accepted or not, is logged along with its source, destination, type of service, and action taken. The Connection Manager also allows control over the maximum number of connections that can be simultaneously active for each service, as well as the maximum rate at which connections for each service are processed.

## The FTP Proxy

The FTP proxy server handles connection requests on the FTP port. Connections that originate inside Interceptor are normally allowed to use the proxy transparently, while connections that originate outside Interceptor usually must provide special authentication.

All connection attempts are logged. For transparent connections, the FTP proxy is invisible to the client. For connections requiring authentication, the user must enter a user name and password when the FTP proxy requests one, and then initiate a second login sequence to instruct the proxy to connect to the FTP server.

Once a connection is established, the FTP proxy relays traffic between the client and the remote server, while at the same time it monitors and controls the commands being sent. Specific FTP commands can be disabled or logged based on the access policy that applies to the connection. The proxy supports both normal and passive mode data transfers with clients, and can be configured (by the access policy) to initiate either normal or passive mode data transfers to the server.

Application proxies are a trusted delivery mechanism, protecting your network from external invasion. When an application requests a connection through the firewall, Interceptor intercepts and verifies the connection requested by the service. If approved, it establishes a separate connection to complete the task. Because the proxy is a trusted delivery mechanism, the outside service is never in direct contact with your organization's network, or with its valuable data assets. Figure 14.110 illustrates this concept on Interceptor.

## Telnet and Rlogin Proxy

The Telnet and Rlogin proxy servers handle connection requests on the Telnet and Rlogin clients respectively. Connections which originate inside Interceptor are normally allowed to use the proxy transparently, while connections which originate outside Interceptor are usually required to provide special authentication. All connection attempts are logged.

For transparent connections, the proxy is invisible to the client. Otherwise, the proxy prompts the user to enter authentication information and then a destination host. Once a connection is established, the proxy relays traffic between the client and the remote host.

## HTTP Proxy

The HTTP proxy server handles connection requests on the HTTP port. It allows internal web browsers to access remote HTTP and FTP servers. It also supports the relaying of SSL-encrypted connections with secure HTTP and NNTP servers.

## E-Mail Proxy

All e-mail between the internal protected network and the external Internet is handled by the Interceptor host. Secure handling of e-mail through Interceptor host is achieved using a two-step process.

First, all SMTP connections to Interceptor are answered by the SMTP proxy program which runs without privileges and simply receives the incoming message, checks if it is allowed by the access policy, and if so hands it off to the sendmail

program which performs the final delivery. The benefit of this approach is that malicious clients never speak directly to the sendmail program and thus cannot exploit any weaknesses it can contain. Instead they interact with a bare-bones SMTP server program small enough to be inspected and verified.

## **X11 Proxy and Generic TCP Proxy**

The X11 proxy allows X Window-based GUI applications (X clients) running on one side of Interceptor to display their output on an X server on the other side. A typical use of this proxy server is to allow an internal user to invoke a GUI program on an external host and display its output on the user's local desktop.

The generic TCP proxy handles a variety of services such as NNTP, Whois, Gopher, Finger, POP, CompuServe and AOL.

## **The Authentication Server**

The Authentication Server programs are an extended optional feature of the Interceptor Firewall System. They support enhanced user authentication for the Telnet, Rlogin and FTP proxy servers. A number of enhanced authentication mechanisms are supported, including the SecurID card from Security Dynamics.

## **The Domain Name Service**

The Interceptor host can be registered with the Internet NIC as the primary name server for your domain. It provides information to the Internet about only the portion of your network that is externally visible.

In most cases, it is just the Interceptor host itself. In addition, it provides Mail Exchange records to direct all incoming e-mail for your domain to the Interceptor host.

## **Real Audio/Real Video Proxy**

This proxy handles the Real Audio/Real Video Protocol and allows transmission of real audio sound files and real video files through Interceptor.

There is a proxy to handle the VDOLive protocol and allows the transmission of VDOLive video files through the Interceptor firewall.

## **RADAR and Utility Command Server**

The Remote Administration Diagnostics and Reporting (RADAR) Server provides a facility for secure, remote administration of Interceptor via a World Wide Web browser.

The Utility Command Server allow user to initiate X11 proxies and ping and trace route diagnostic utilities via a World Wide Web browser.

## **Web Caching and Java and ActiveX Blocking**

With Interceptor version 3.0, you can set up a web cache on the firewall system. If there are many internal people who request the same outside resource (and there usually are), this feature will retrieve the information only once and store it on the firewall. This feature greatly increases performance for most organizations.

Also, each time you retrieve a WWW page, the browser makes a new and separate connection for the text and every image contained on the page. This feature allows the connection to stay open until all the information is retrieved, therefore significantly increasing performance of WWW activity.

Technologic has incorporated Java and ActiveX applet filtering into Interceptor.

## Multiple Firewall Management

This feature helps people manage increasingly complex Internet usage. Interceptor allows you to set up multiple firewalls in groups. Through RADAR, you can manage these groups. You make one change and RADAR updates all the firewalls in the group at once. This allows you to maximize the security expertise in your organization and provide concise, consistent Internet access policy.

## Systems Requirements

Interceptor Firewall requires:

- Intel-based Systems Pentium 90Mhz
- 16 MB RAM
- 500+ MB Fast SCSI-2 Hard Disk Drive
- 500+ MB SCSI Tape Drive
- Two Ethernet or Token Ring Network Adapters
- Standard VGA Video Card and Monitor

## Sun's Sunscreen EFS Firewall - a Stateful Inspection Firewall

With world headquarters in Mountain View, Calif., Sun Microsystems, Inc., has been described as a "full service provider that can compete on an equal footing with IBM and Hewlett-Packard Co." (InformationWeek, Feb. 13, 1995). The company was founded in 1982 on the premise that "the network is the computer." This simple, yet revolutionary concept helped change the face of the computer industry and has propelled the company into a thriving \$6 billion company.

While the company's legacy has been as a technical workstation supplier, Sun is successfully transforming itself into an enterprise computing firm focused on global network computing. Sun believes that the vast network and resources that exist beyond a person's own computer is where the true strength of information technology lies. Unlike PCs -- which were built to enhance individual productivity -- workstations incorporate networking into its design core to allow groups of people to collaborate, thereby improving company-wide productivity.

Nonetheless, to meet the rapidly evolving needs of today's networks, corporations require an integrated security solution that is flexible and scaleable. Sun has created a suite of security solutions that scales to meet enterprise needs: the SunScreen suite.

Figure 14.111 is a screenshot of Sun's Web site.

### Note:

For more information, contact Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, CA 94303, telephone at 1-800-SUN-FIND or 1-972-788-3150 outside the United States. You can also contact them via e-mail at [sunscreen@incog.com](mailto:sunscreen@incog.com) or at their Web site at URL <http://www.sun.com>.

## The SunScreen Model

With a mission to provide the enabling products, services and technologies for secure electronic commerce and communication over public networks, Sun leads the evolving market in infrastructure and architecture. SunScreen products provide the foundation for secure Internet access and electronic commerce. The SunScreen product line focuses on enabling corporations to create secure virtual private networks (SVPNs) and provide network access control.

The traditional way of securing corporate networks, as shown on figure 14.112, has been with firewall based perimeter

security, separating the networks into static safe and unsafe areas much like creating fences. The problem with this approach is that once the fence has been breached, the network can be compromised.

Sun's solution for this problem is a suite of products, which includes,

- **SunScreen SPF**, a dedicated, stealthy, network security solution, designed for the highest security needs of complex networks; typically deployed at the gateway to a public network;
- **SunScreen EFS**, an encryption server software product with strong firewall/gateway functionality. It can be used to protect all servers in the de-militarized zone (e.g. FTP, WWW, mail) and Intranet (e.g. database, HR, payroll servers); and
- **SunScreen SKIP**, which provides encryption and key management capabilities, to the desktop or remote end user, which enables PCs, workstations, and servers to achieve secure/authenticated communication.

Figure 14.113 illustrates the SunScreen line and how they fit in your security policy.

Sun security implementation vision is scaled to enterprise needs as secure virtual private networks are deployed in volume, as shown on figure 14.114.

The SunScreen product line enables you to secure your network in an entirely new way. SunScreen SPF provide stealthy network access control and SVPN solutions. SunScreen EFS provides similar network access control and encryption capability, allowing corporations to lock down each of the DMZ machines, as well as all the servers within the corporate network. This secures the whole network, not just the perimeter.

Deploying the product line will help create multiple SVPN's both within the Intranet and Internet environments. Each department, from the corporate office, to finance, to personnel, can each have a separate secure network. Secure and authenticated communication with remote customers and employees, as well as business-to-business communication can be accomplished via SunScreen SKIP.

This creates a network security system involving dedicated gateway-level security with SunScreen SPF a hardened encryption server for databases, NFS, mail, Web and other types of application machines with SunScreen EFS and encryption equipped end nodes with SunScreen SKIP.

This secure network solution creates one large electronic workspace, as shown on figure 114, where distinctions between Intranets and the Internet become academic from a security standpoint, and all communication can be made private and authenticated as needed.

Sun's SKIP technology allows you to use the Internet as a conduit to your business partners and employees. According to Sun, studies have shown that this can reduce the overall operating expenses by 23% (U.S. Computer).

## ***Secure access control.***

By using stateful, dynamic, packet screening and rules-based technology, SunScreen products allow filtering at the packet level while retaining application-level intelligence. Packets are examined based on filtering rules, and are completely customizable. They may be filtered by connection type, source or destination address, protocol, or protocol port number, in addition to user-definable services. You determine which hosts are granted access to your network, when and what types of access are permitted, and what constitutes a security violation.

Also, because the encryption of data occurs at the network (or "IP" layer), existing applications do not require modification to take advantage of the SunScreen product family's privacy features. In fact, all existing TCP/IP-based applications immediately reap the benefits of SKIP encryption and key management when any SunScreen product is installed.

## ***Ease of administration.***

Combined with a user-friendly interface and centralized control, SunScreen products allow for ease of maintenance and management with little training and low software maintenance costs. Web-based administration also allows for flexibility in selecting the number and placement of administration stations.

SunScreen products provide centralized and granular control of all authenticated users. SKIP authenticates remote clients for secure communication between an enterprise's local network and the corporate branch offices, business partners, and nomadic users. Remote access can be granted or denied using a number of criteria such as network address or key identifier in the case of nomadic systems. Figure 14.115 and 14.116 shows an example of such a scenario.

## ***SunScreen SPF-200 and SunScreen EFS Security Solutions***

SunScreen SPF-200 is Sun's premier security platform for perimeter defense and electronic commerce. SunScreen EFS complements SPF as the platform to secure all departments and sites within an organization. Together, they protect the entire organization, securing electronic commerce, remote access, and Extranets.

The SunScreen SPF-200 security solution is the premier perimeter defense in the industry. Its strength is in stealthing: no IP address is seen. Stealthing makes SunScreen SPF essentially impenetrable from the Internet because an intruder cannot address the machine. The SunScreen SPF product also scales to almost whatever level is required and supports high-speed, secure communication over the Internet.

SunScreen EFS software is designed for wide-spread deployment within a firm to protect key departments and sites, as well as for deployment of multiple Extranets. It is a powerful combination of a high-performance encryption server along with a strong firewall.

### **SunScreen SPF's Features**

SunScreen SPF package offers a set of solutions to your company's security as outlined below:

- Stealthing to help protect an organization from Internet attacks.
- Top performing perimeter defense to screen a high level of Internet traffic.
- A multithreaded encryption engine to meet high-end electronic commerce requirements.
- State-of-the-art SKIP encryption to enable secure electronic commerce and remote access for employees.
- Remote administration.

### **SunScreen EFS' Features**

SunScreen EFS package offers a set of solutions to your company's security as outlined below:

- High-speed dynamic packet screen.
- A multithreaded encryption engine to meet high-end Extranet requirements.
- State-of-the-art SKIP encryption.
- Remote administration.

## ***SunScreen SPF-200***

The SunScreen SPF package is Sun's strategic platform for perimeter defense, providing secure business operations over the Internet. To ensure a high level of security, SunScreen SPF uses a stealth design to protect it from attack, and state-of-the-art SKIP encryption to protect data going over the network. Its advanced dynamic packet filtering, coupled with Sun's high-speed hardware, is designed to meet the most demanding performance requirements. The SunScreen SPF solution enables organizations to deploy a premier perimeter defense today, and accommodate business over the Internet at their own rate in the future

### **Features and Benefits**

The following are the key features of SPF-200:

- **Top performing perimeter defense** - According to Data Communication magazine (March 21, 1997, SunScreen EFS was the fastest firewall among the top firewall products available on the market. Given SunScreen SPF's internal



design and optimization, SunScreen SPF should run even faster. SunScreen SPF performance ensures that it can keep up with the demands required to screen large amounts of Internet traffic.

- **The stealth design** - This design makes SunScreen SPF not addressable with an IP address provides two benefits. First, stealthing makes a SunScreen SPF system more secure because potential intruders can not address the machine running SunScreen SPF, possibly compromising the machine. Second, installation of SunScreen SPF into the network is easy since the administrator can install it without changing routing tables.
- **The stealth design "hardens" the OS** - This factor turns the system into a dedicated SunScreen SPF system that only runs SunScreen SPF. Hardening the OS enhances security. Since other applications do not run on the system, there is less exposure. SunScreen SPF systems use a separate administration station that can be any SPARC machine and need not be dedicated.
- **State-of-the-art SKIP encryption technology** - This encryption technology provides secure network communication and acts as the infrastructure for electronic commerce, Extranets, and secure remote access. SKIP protects the data being transmitted, ensures its integrity (not altered), and provides a high level of authentication.
- **SunScreen SPF covers both TCP and UDP services** - In regards to UDP, SunScreen SPF maintains state to improve security and performance.
- **SunScreen SPF allows flexibility in logging what has passed or failed through the screen.** - Administrators can choose what they want to monitor and be alerted to problems through pagers or alerts to network management stations.
- **Network Address Translation (NAT) converts internal addresses to a different set of public addresses.** This allows for additional protection for the internal network, and also helps those sites that have not registered their IP addresses. NAT supports both static and dynamic translation of internal addresses to public addresses. Since hackers do not know the internal addresses of hosts, attacks are minimized.
- **Administration is done through secured, remote administration stations** - This enhances the security and meets the needs of organizations for remote management

## SunScreen EFS

SunScreen EFS software is Sun's strategic offering for compartmentalization, where companies deploy multiple screens to protect various departments and sites. SunScreen SPF is the best offering for protecting the corporation from Internet attack and for performing business over the Internet.

In contrast, SunScreen EFS was designed from the ground up to be deployed throughout an organization and protect sites and multiple departments inside the organization. With it, organizations can implement security policy and establish secure connections between departments, sites, or even between business partners over an Extranet.

## Features and Benefits

The following are some of the key features and benefits of SunScreen EFS:

- **High-speed dynamic packet screen** - As mentioned earlier, this firewall was rated the fastest firewall by Data Communication's performance test among the top firewall vendors. SunScreen EFS can meet the performance needs of most any department or site.
- **SunScreen EFS runs on Solaris systems as a separate application along with other applications.** This allows it to be deployed throughout the organization. In contrast, SunScreen SPF stealthing provides the ultimate in security in high-risk areas such as perimeter defense on the Internet.
- **State-of-the-art SKIP encryption technology** - As with SPF, this feature provides secure network communication and acts as the infrastructure for communication between departments, remote sites, and partners. SKIP protects the data being transmitted, ensures its integrity, and provides a high level of authentication.
- **SunScreen EFS covers both TCP and UDP services.** In regards to UDP, SunScreen EFS maintains state to improve performance.
- **SunScreen EFS allows flexibility in logging what has passed or failed through the screen.** The administrator can choose what they want to monitor and also be alerted to problems through pagers or alerts to network management stations.

- **SunScreen EFS can be managed remotely** - This feature makes it very practical to deploy numerous SunScreen EFS servers throughout an organization and manage them centrally.
- **Conversion tool to migrate from Solstice FireWall-1** - This conversion facility translates host group definitions, network object definitions, service definitions, actions, and rules from FireWall-1 3.0 to SunScreen EFS 1.1.
- **Network Address Translation (NAT) converts internal addresses to a different set of public addresses.** - This provide additional protection for the internal network, and also helps those sites that have not registered their IP addresses. NAT supports both static and dynamic translation of internal addresses to public addresses. Since hackers do not know the internal addresses of hosts, attacks are minimized.

## System Requirements

SunScreen SPF-200's stealth feature dedicates the system running the screen to just SunScreen SPF. In addition, SunScreen SPF requires a separate administration station, but is not required to be a dedicated system.

In contrast SunScreen EFS runs as a separate application on any SPARC machine.

The system requirements for the SunScreen SPF-200 Screen are:

- **CPU:** Ultra 1or Ultra 2 or a SunScreen SPF-100 screen for upgrades
- **Disk:** 1 GB of disk
- **Memory:** 16 MB

The system requirements for the SunScreen SPF-200 Administration Station are:

- **CPU:** SPARC system or compatible
- **Operating System:** Solaris 2.4, 2.5, or 2.5.1
- **Disk:** 100 MB of free disk space
- **Memory:** 16 MB

As for the SunScreen EFS, the system requirements are:

- **CPU:** SPARC system or compatible
- **Operating System:** Solaris 2.4, 2.5, or 2.5.1
- **Disk:** 100 MB of free disk space
- **Memory:** 16 MB

## Solstice FireWall-1 3.0

Another firewall product offered by Sun which deserves to be mentioned is the Solstice FireWall-1 software, which provides Internet and Intranet data security for the enterprise network in a distributed environment on Solaris and Windows NT platforms.

Solstice FireWall-1 Version 3.0 is one of the leading network security system for creating and managing TCP/IP firewalls. Solstice FireWall-1 software enables an enterprise to build its own customized security policy, yet is installed and managed from a single workstation console. As an enterprise firewall solution, Solstice FireWall-1 3.0 has the flexibility, scalability, extensibility, and cross-platform support to meet a company's security needs.

### Solstice FireWall-1 Features

Solstice FireWall-1 is based on Stateful Multi-Layer Inspection technology, delivering superior security, connectivity, and performance. It offers excellent network and application-level security, along with user authentication, for virtually any size enterprise, enabling safe access to the Internet's vast resources. This technology delivers a superior solution compared to competitors' products that are based only on application gateways, proxies, or simple packet-filtering.

Installed on a gateway server, the Solstice FireWall-1 inspection module acts as a security router for traffic passing between a

company's Intranet segments or between the internal network and the Internet. All inbound and outbound data packets are inspected, verifying compliance with the enterprise security policy. Packets that the security policy does not permit are immediately logged and dropped.

## Comprehensive Services Support

By incorporating dynamic, application-level filtering capabilities and advanced authentication capabilities, Solstice FireWall-1 enables true connectivity for over 120 built-in services, including secure Web browsers and HTTP servers, FTP, RCP, all UDP applications, Oracle SQL\*Net and Sybase SQL Server database access, RealAudio, Internet Phone, and many others.

Solstice FireWall-1 runs on the Solaris operating environment for SPARC and Intel platforms as well as on Windows NT for Intel platforms.

A management module running on one platform can manage inspection modules running on other supported platforms. The management module itself is now a client/server application, with a GUI client that runs on Windows 95 and Windows NT, as well as on all supported platforms.

## Encryption Support for Data Privacy - Virtual Private Networks

The Solstice FireWall-1 encryption module enables virtual private networks and commerce over the Internet by encrypting all traffic over the Internet. It uses a highly efficient "in-place" encryption. By maintaining the size of the encrypted data packets, communications lengths are not altered and packet fragmentation is eliminated. The highest network performance is achieved, and routing priorities and policies are preserved.

Another important feature is the so called SecuRemote feature, which creates a virtual private network for Windows 95 and Windows NT users connecting to their networks with dial-up connections over the Internet or the public switched phone network to any Solstice FireWall-1 system running the optional VPN or DES encryption. SecuRemote will transparently encrypt any TCP/IP-based application, without change to the application itself.

Solstice FireWall-1 also supports the SKIP protocol, which was invented by Sun. This allows Solstice FireWall-1 installations to create a virtual private network with any other products, from Sun and other vendors, remaining compatible with industry standard.

## Client Authentication

Solstice FireWall-1 provides centralized and granular control of all users, including authenticated and unknown users. Client Authentication permits only specified users to gain access to the internal network, or to selected services, as an additional part of secure communications between an enterprise's local network and corporate branch offices, business partners, and nomadic users. Client Authentication works without modifying the application either on the client or server side.

This firewall supports four different approaches for user authentication, including Security Dynamics' SecurID one-time password cards. Unknown users can be granted access to specific services such as Web servers or e-mail, depending on your corporate security policy.

This firewall can protect users from viruses and malicious programs that enter a company's network from the Internet. This includes viruses in executable programs, "macros" that are part of application documents, and ActiveX and Java applets. It also uses third-party "plug-in" anti-virus and URL-filtering programs available from such vendors as Symantec, McAfee, Trend Micro, Cheyenne, Eliashim, WEBsense, and others.

If you are operating a "server farm," Solstice FireWall-1 can optionally distribute incoming requests to the next available server. One logical IP address can support access to all servers.

## Anti-Spoofing and SNMP Management

Spoofing is a commonly used technique to gain access to a network from outside, the Internet for example, by making packets appear to come from inside the network or firewall. Solstice FireWall-1 detects such packets and drops them, and can also log and issue an alert.

As for SNMP management. Solstice FireWall-1 has version 2 SNMP agents that integrate it to Solstice Domain Manager, Solstice Enterprise Manager, or other enterprise management tools.

## Secure Computing's Borderware Firewall: Combining Packet Filters and Circuit-Level Gateways

Headquartered in St. Paul, Minn., Secure Computing is one of the largest network security companies in the world. Secure Computing's services and comprehensive suite of interoperable products address every aspect of enterprise network security including consulting services, firewalls, Internet monitoring and filtering, identification, authentication, authorization, accounting and encryption technologies. The only network security company that provides end-to-end network solutions encompassing all universal enterprise security standards, Secure Computing has more than 4,000 customers worldwide, ranging from small businesses to Fortune 500 companies and government agencies.

Figure 14.117 is a screenshot of Secure Computing's Web site.

**Note:**

For more information, contact Secure Computing at 2675 Long Lake Road, Roseville, MN 55113. Tel +1.612.628.2700 Fax +1.612.628.2701. Or via e-mail at [sales@securecomputing.com](mailto:sales@securecomputing.com) or via the Web at <http://www.securecomputing.com>.

## *The BorderWare Firewall Server*

The BorderWare Firewall Server defines a new product category of firewalls by combining packet filters and circuit-level gateways with application servers into a single, highly secure, self-contained system. It is a powerful, advanced security product that protects TCP/IP networks from unwanted external access as well as provides control of internal access to external services.

Using the BorderWare Firewall Server you can connect your private TCP/IP network to the global Internet, or to other external TCP/IP networks, and remain confident that unauthorized users cannot gain access to systems or files on your private network. Figure 14.118 shows a typical layout of BorderWare Firewall configuration

The benefits and uses of BorderWare go far beyond protecting your network from external access as this firewall provides many other services, including a secure Mail server, dual Name servers (internal and external), a News server, an anonymous FTP server, a WWW server and a Finger information server.

The BorderWare Firewall Server is transparent to your internal network users. This means that all of the TCP/IP networking applications that your organization currently uses, including DOS and Windows driven software, will continue to work without modification.

The creators of BorderWare kept two things in mind when designing the firewall server: simplicity and security. It is simple as a light switch, but just turn it on and you can cross a threshold to the most complete set of features available to a firewall.

BorderWare has a simple graphical user interface (GUI) for configuration, setup, and control of the firewall server, which saves you from learning access rules syntax or the proper order that they must be defined. BorderWare lets you configure all aspects of the firewall through the GUI. DNS, Mail, News, outbound access, WWW, FTP, and alarms are just a few examples. BorderWare even lets you enable your own user-defined services, in an absolutely secure manner.

As discussed on chapter 7, "What is an Internet/Intranet Firewall After All?," firewalls come in three types: packet filters, circuit-level gateways, and application gateways. BorderWare combines all three into one firewall server giving you the flexibility and security you need, as seen on figure 14.119. BorderWare also supports multiple styles of authentication including address/port based authentication and cryptographic authentication.

There is one very important feature of BorderWare that really stands out in the crowd of commercial firewalls on the market today. BorderWare is built from the bottom up with a fail-safe design. The foundation for BorderWare was a securely hardened kernel. Each layer of functionality that was added was first made secure. In the event that any of these services is under attack, the firewall is still not compromised. There are tiny firewalls inside BorderWare that keep barriers around the services to prevent the spread of any compromised piece and the rest of the firewall remains unaffected.

The following is a list of the main features found on BorderWare:

- **Easy to use** - It works with any PC, MAC or UNIX Internet application and offers complete transparency to internal users. There is no need to change application software or user procedures.
- **Has all you need to link to the Internet** - , Enables you to incorporate application servers like Mail, News, WWW, FTP, DNS
- **Makes joining the Internet easy** - It remaps and hides all internal IP addresses, allowing use of non-registered IP address
- **Is a complete network security solution** - It combines packet filtering with application-level and circuit-level gateways.
- **Provides worry-free inbound access** - It permits authenticated inbound Telnet access using one-time password "tokens".
- **Is flexible** - It allows the security administrator to define proxies for secure and specialized applications that require "tunneling" through the firewall.
- **Is easy to install and manage** - It provides a simple graphical interface for configuration, control and set-up.
- **Lets the administrator know then the system is being attacked** - It incorporates security features to detect probing and initiate alarms.
- **Makes audit simple and foolproof** - It includes comprehensive audit capabilities and allows the security administrator to direct log files to a remote host.

## Transparency

BorderWare provides outbound application services such as Telnet, FTP, WWW, Gopher and America On Line transparently. Existing windows-based or non-windows-based point-and-click client software will run without modification. You can use your favorite shrink-wrap software. There is no need to login to the firewall. BorderWare is transparent.

## Network Address Translation

BorderWare remaps and hides all internal IP addresses. The source IP addresses are written so that outgoing packets originate from the firewall. The result is that all of your internal IP addresses are hidden from the users on the Internet. This gives you the important option of using non- registered IP addresses on your internal network. In some cases this saves users hundreds of hours of work.

## Packet Filtering

All IP packets going between the internal network and the external network must pass through BorderWare . User definable rules allow or disallow packets to be passed. The graphical user interface allows system administrators the ability to implement packet filter rules easily and accurately.

## Circuit-Level Gateway

All of outgoing connections and incoming connections are circuit-level connections. The circuit connection is made automatically and transparently. BorderWare allows you to enable a variety of these such as outgoing Telnet, FTP, WWW, Gopher, American On Line, and your own user-defined applications. Incoming circuit-level applications include Telnet and FTP. Incoming connections are only permitted with authenticated inbound access using one-time password tokens.

## Applications Servers

One of the extra features of BorderWare is that it includes support for several standard application servers. These include: Mail, News, WWW, FTP, and DNS. Each application is compartmentalized from other firewall software, so that if an individual server is under attack, other servers/functions are not affected.

## Audit Trails and Alarms

BorderWare has comprehensive audit and logging capability. It also provides alarms when probing is detected.

Log files are kept for all connection requests and server activity. The files can be viewed from the console displaying the most recent entries and scrolls in real time as new entries come in, as seen on figure 14.120. These files can also be retrieved from the firewall using the administrative FTP user from your internal network.

The log files include:

- connection requests
- mail log file
- news log file
- other servers
- outbound FTP sessions
- alarm conditions
- administrative log
- kernel messages

Log information that is sent to the FTP log area can now be sent to another internal machine running syslog. Also, BorderWare has an alarm system that watches for network probes. The alarm system can be configured to watch for TCP or UDP probes from either the external or internal networks. Alarms can be configured to trigger email, pop-up windows, messages sent to a local printer, and/or halt the system.

## Transparent Proxies

Traditional firewalls require either logging into the firewall system or the modification of client applications using library routines such as "SOCKS". BorderWare permits "off-the-shelf" software such as Beame & Whiteside BW-Connect TCP/IP package, NetManage Chameleon, SPRY AIR Series, and standard UNIX networking software to operate transparently through the firewall. Figure 14.121 shows the many protocols BorderWare incorporates proxies to.

Integrated Servers BorderWare includes support for several standard applications including Mail, News, FTP, Finger, Name Server (DNS), and WWW. Each applications is completely isolated from all other applications, so that attempts to compromise one server can have no effect on the others.

## *BorderWare Application Services*

The BorderWare Firewall Server, as seen on figure 14.122, incorporates two separate DNS servers on the firewall itself:

- **External DNS** - The External DNS server provides a limited external view of the organizational domain and initially configures itself with a number of standard names that all point to the firewall itself (such as Mail, News, FTP, NS and WWW). It also has specific entries for the domain so that connections can be conveniently made using only the

organizational domain name and whatever additional hostname is specified for the firewall. The External DNS also automatically installs NS and wildcard MX records that point to the firewall. Additional backup MX and secondary NS records can be configured by the administrator. No internal information is available to the External DNS and only the External DNS can communicate with the outside. Therefore, no internal naming information can be obtained by anyone on the outside. The External DNS cannot query the Internal DNS or any other DNS inside the firewall.

- **Internal DNS:** The Internal DNS is automatically configured with some initial information and can have additional hosts added via the administrator interface. Other internal domains or sub-domains can be primaried, secondaried or delegated to other internal nameservers. The ability to prime the internal DNS by downloading host and NS delegation information from an existing DNS is available in the next major release. The information managed by the Internal DNS is only available to internal machines. The Internal nameserver cannot receive queries from external hosts since it cannot communicate directly with the external network. Resolution of external DNS information both for the firewall itself and to handle internal queries for external information are handled by the internal nameserver. Although it is unable to communicate directly with the external network, it is able to send queries and receive the responses via the External DNS.

## Mail Servers (SMTP and POP)

The BorderWare mail system was originally designed with a security model in mind, as shown on figure 14.123. It is based on ZMailer, a mature mail system in use on major Internet gateways. The author has made further specific enhancements for the BorderWare product.

The system consists of independent programs for SMTP reception, routing decisions, SMTP delivery, delivery scheduling, and other work. ZMailer has no code relation to Sendmail and has not in the past been susceptible to any of the security problems with Sendmail. In this product it also runs without special privileges in an isolated environment.

The BorderWare mail system can act as a corporate Internet or SMTP mail gateway. It allows the administrator to explicitly specify mail routing information so that a subversion of DNS data cannot be used to hijack mail. It is also an example of how the two-faced nature of the BorderWare system extends into application-level functionality. The mail system can easily be configured to completely hide the structure of an internal mail environment from the outside world without the inside users being aware of this. It is capable of arbitrarily mapping from internal addresses to external addresses, as may be desired due to either information leakage or corporate image considerations. The virtual division of views is carried to the point of foiling external email probe attempts, and manipulating outgoing message headers to remove any internal naming information that would otherwise be leaked.

## Mail Domain Name Hiding

With BorderWare, if you ever decide to map several internal subdomains to a single organizational external domain at your company, the potential conflicts due to non-unique user ids can be resolved automatically by the mail system in its "training" mode. When that feature is enabled, new internal email addresses that arrive on the BorderWare Firewall are translated into unique externally visible addresses. If desired you can later disable the auto-creation feature and begin exercising manual control over the mappings. This allows an easy introduction of this kind of control into an existing gatewayed environment. It also allows administrative control over access to Internet email on a per-user basis. The administrator is of course always able to explicitly create or delete mappings between internal and external addresses.

## POP Mail Server

In addition, the BorderWare system contains a POP3 server so that it can be used to directly support a typical client/server mail environment that uses commercial PC/Mac-based software. User mailboxes defined on the BorderWare Firewall Server take precedent over the internal message routing information. People within a single internal administrative subdomain can be given the option of whether they prefer to use POP mail or a traditional host-based mail system.

## Anonymous FTP Server

BorderWare incorporates a secured anonymous FTP server which provides read-only access to a protected and limited file hierarchy. The GUI provides a mechanism to enable a writeable incoming directory to allow the sending of files to the firewall. An administrative account, only accessible from the internal network, is the single method of accessing and maintaining the data areas.

## News Server

BorderWare incorporates a secured and self maintaining NNTP based news server. It accepts an Internet news feed from designated external systems, usually your Internet service providers news machine(s). The news can be read directly from BorderWare with standard PC or UNIX news reader clients. Also, the news can be fed to internal or external sites. No maintenance is required for the news server as there is auto-addition of new News-groups and auto-deletion of old News.

## Web Server

BorderWare incorporates a secured HTTP server. It will respond to internal or external requests for files from a limited file hierarchy. Internal users will be transparently proxied to other Internet WWW servers. However, external users will never be able to access any WWW server running on the internal network.

## Finger (Information) Server

Finger is a standard utility that can be used for probing systems and it is useful to know who is examining your system. The BorderWare finger (information) server will respond to a request by displaying a customizable file. This file usually contains static information about your company such as phone numbers and addresses. The full request is logged.

## Encryption Features

Using a DES encryption based electronic challenge and response authentication card you can Telnet or FTP to the internal network from an external network. As soon as you request a Telnet or FTP session, you are prompted with an eight digit challenge number. The next Telnet or FTP attempt would be given a different challenge and would require a different response.

## Automatic Backups

BorderWare has a built-in mechanism for automatic nightly backup. First, it does a backup of your configuration files onto a floppy diskette. It also backs up all your anonymous FTP directories, WWW data, and Finger server data on 4mm DAT tape. News data is not backed up for obvious reasons because of the amount of space it would use. When upgrading your software, you simply restore your configuration from the diskette and restore your data files from tape in minutes. The backup is also very useful if your system crashes due to any hardware failure.

## Security Features

The BorderWare Firewall Server is unique in integrating secure application servers as part of the basic system. Each server has been designed from the ground up with security in mind. This alleviates the necessity for you to modify and harden your own server application or machine as is required by some firewalls. Figure 14.124 gives an overview of the Secure Server Net (SSN).

The BorderWare Firewall Server is built upon a version of UNIX that has been hardened to protect against security violations. The operating system has been modified so that even if an attacker did gain access to the firewall through a service s/he would be unable to affect the other application systems or gain access to your internal network.

The BorderWare Firewall Server has secure versions of most Internet services and networking tools including:

- a dual Domain Name Servers (internal and external)



- a secure SMTP server
- a secure anonymous FTP server
- a secure World Wide Web server
- a secure Finger Information server

A variety of mechanisms are used to further enhance the integrity of the BorderWare Firewall Server and protect the internal network from unauthorized access such as:

- internal IP addresses are hidden so all internally originated traffic appears to come from the firewall itself
- lures and other mechanisms to detect probing from Internet
- challenge/response authenticated inbound Telnet access
- alarms triggered from external/internal probes
- file integrity checking to protect from subversion of the firewall software

## Ukiah Software's NetRoad Firewall: a Multi-Level Architecture Firewall

Ukiah Software is a Silicon Valley-based developer of Internet and Intranet software products. Their mission is to deliver solutions for secure information access over the Internet and Intranet, in environments requiring multi-platform and multi-protocol support. Ukiah is the only company offering advanced firewall products for heterogeneous TCP/IP and IPX/SPX environments, running on NetWare and Windows NT. Their firewall is also one of the most manageable firewall on the market today, through its integration with NDS.

Ukiah's flagship product, NetRoad Firewall, delivers advanced multi-level security incorporating application level gateway, circuit level gateway and packet level filtering functionality. This multi-level architecture delivers the highest level of firewall security in repelling an extremely broad array of security attacks.

Figure 14.125 is a screenshot of Ukiah's Web site.

### Note:

For more information, contact Ukiah Software, 2155 South Bascom Avenue, Suite 210, Campbell, CA 95008, (800) 988-5424 or (800) 98-UKIAH, Fax: (408) 369-2899. Via e-Mail, [sales@ukiahsoft.com](mailto:sales@ukiahsoft.com) or on the Web URL: <http://www.ukiahsoft.com>.

## *NetRoad FireWall for Windows NT and NetWare*

NetRoad FireWall, as seen on figure 14.126, provides multi-level firewall security and network address translation for TCP/IP as well as IPX clients. It is the only one of its kind on the market today. Runs integrated with Windows NT Domains, or can be integrated with Novell's NDS. For NetWare, NetRoad FireWALL is directly integrated with NDS.

Ukiah Software's NetRoad FireWall provides a security firewall for both your TCP/IP and your IPX clients. By combining in a single, integrated product the capabilities of both an IP firewall and an IPX/IP gateway, FireWall delivers seamless security for mixed protocol networks. No need for separate firewalls and IPX/IP gateways. And, FireWall allows you to secure your entire network in an easily managed way, from a single management console integrated into NDS or other LDAP-compliant directory service.

The following is a list of the key features of NetRoad:

- Two products in one--an IP firewall and an IPX/IP gateway
- First-class IP firewall security for Internet and Intranet connections
- Secure IP connectivity for both IP and IPX clients

- Integrated with Novell Directory Services (NDS) and Windows-based management
- Available for NetWare and NT

## Security for Mixed Protocol (IP and IPX) Networks

With mixed IP and IPX protocol networks now the norm in most organizations, a firewall must offer Internet connectivity and security to both. NetRoad FireWALL provides an IP firewall and Network Address Translation (NAT) to hide your internal addresses from the Internet. It also transparently controls the full range of TCP/IP operating systems and applications.

Various alternatives to providing security in mixed protocol networks do exist, but all represent only a fraction of the total required solution:

IPX/IP gateways provide Internet connectivity for IPX clients, but the security is very basic. Application security, for example, is generally based only on TCP ports - some products also support ICMP or UDP port-based filtering. The security focus with most of these gateway products is on controlling outbound access, not on dealing with the more serious problem of inbound network access. More importantly, these gateways do nothing to provide security for IP clients.

Filtering bridges or packet-level filtering by routers are partial solutions, but they also have major security limitations (see section on Types of Firewalls), and don't support Internet services for IPX clients.

IP firewalls can provide great security (as long as they provide capabilities up to and including an application level gateway), but they only support IP clients - not IPX.

Dual-protocol-stack clients can be implemented to get around the IP-only nature of the Internet and of IP firewalls, but this is complex to implement and manage and is likely to be a nightmare for network administrators.

Only NetRoad provides a true firewall today running on NetWare, and also on any platform - NetWare, Windows NT or other - that provides both an integrated IP firewall and an IPX/IP gateway. No other product family offers integrated firewall support for both IP and IPX clients, and also offers this firewall on both NetWare and Windows NT servers. Figure 14.127 illustrates this concept.

As previously discussed, the most secure form of firewall is a 'multi-level firewall' - one which combines packet filtering, a circuit-level gateway, and an application-level gateway firewall to provide defense in depth (see figure 14.128). Since security attacks can and will come at any level that exposes security vulnerabilities, the combination of multiple levels of security is the only way to have a fighting chance against the determined attacker.

For IPX clients, FireWALL provides an IPX/IP gateway, supporting any Winsock 1.1 compliant TCP-based application such as Web browsers, FTP and Telnet, as well as UDP applications such as RealAudio, most real-time services, DNS, and SNMP.

It also supports ICMP-based ping. It's simple and inexpensive to install, since it doesn't require any changes to IP or IPX stacks, and you don't have to install or manage a TCP/IP stack on IPX clients!

## Simple Management and NDS Integration

FireWALL integrates with Novell Directory Services (NDS). According to the vendor, LDAP-based directory service is soon to be released. This means that FireWALL can execute policies based on users that have already been defined! Additionally, all FireWALL's configuration information can be maintained in a single repository, or replicated across multiple repositories for greater fault tolerance. Besides offering NDS integration, FireWALL for Windows NT can also be managed on a stand-alone basis, without a directory service.

Other management features include:

- Alarms through
- e-mail,
- pager,
- SNMP trap,

- NDS log entry, and
- On-screen messages.
- Statistics that keep tabs on security threats and user activity.
- Remote management, including encryption and authentication, are built into the FireWALL solution.

## Multi-level Firewall Security and User Authentication

FireWALL is a multi-level firewall, enforcing security at the network, circuit and application level. Application level inspection modules are provided for the most common applications: HTTP, FTP, Telnet, SMTP, Real Audio, and so on. This architecture provides the highest level of security against the broadest array of security threats.

The multi-level approach also ensures a very flexible degree of control and security policies can be tailored as precisely as required to control traffic. Network traffic passing through the firewall can be filtered based on the following criteria:

- users,
- destinations,
- groups,
- time of day,
- applications,
- individual application commands, and
- file types and even right down to the level of individual Web pages.

The addition of three different forms of user authentication (NDS and MD4/MD5 One Time Password) make FireWALL a robust security solution.

## *NetWare and NT Firewall Support*

FireWALL runs on Windows NT 4.0, IntranetWare, and NetWare 4.x. A common feature set is implemented on both the NT and NetWare platforms, so that implementing multiple firewalls on different platforms is transparent to the administrator. Both offer common capabilities, and are managed in a common fashion through NDS. This ensures a strong security system in mixed protocol and mixed platform environments. Whether your long term goal is simply co-existence, or migration to a single protocol and platform, FireWALL offers you a choice.

NetRoad FireWALL can be used in a wide variety of network configurations, as seen on figure 14.129, including:

- IPX clients only
- TCP/IP clients only
- Mixed protocol configurations (the most common network configuration)

The platform on which FireWALL runs can be either NetWare 4.x or IntranetWare, or Windows NT. Access from the firewall to the Internet can be provided via a stand-alone router (such as Cisco, Bay Networks etc.) or the multi-protocol routing (MPR) capability in NetWare itself.

## High Performance

A highly efficient application implementation delivers high throughput and hence maximum performance for client applications. With 95% throughput efficiency, FireWALL has the performance edge for Internet and high-speed intranet connections.

# ***Future Evolution of the NetRoad FireWALL Platform***

According to Ukiah, the NetRoad FireWALL platform is designed to be just that: a platform. Its robust design will allow it to continue to evolve over the long term, adding new capabilities through the simple integration of third-party products, such as encryption and user authentication applications, as well as through new features and modules added by Ukiah itself, as shown on figure 14.130.

FireWALL has many advantages that make it singularly well-suited to play the platform role over the long term. Examples of these advantages include:

- Multi-protocol architecture that supports complex networks
- Portability across operating system platforms, both stand-alone and embedded
- Multi-layered security that ensures maximum flexibility to meet the security threats of today and tomorrow
- Integration into directory services and network management platforms that ensures a cohesive, easy to manage system for organizations large and small
- Extensibility of NetRoad FireWALL's policy-based architecture that allows the incorporation of other application modules that add new facets to the platform, beyond network security.

## ***System Requirements***

The following are the requirements for FireWALL Server for NetWare:

- NetWare 4.x or IntranetWare
- Novell's TCP/IP stack
- At least 2 network interface cards
- Pentium 133 or higher
- 20MB free disk space
- 16MB RAM

As for FireWALL Server for Windows NT the requirements are:

- Windows NT 4.0 or later (workstation or server)
- TCP/IP stack
- At least 2 network interface cards
- Pentium 200 or higher
- 20MB free disk space
- 200MB swapfile size
- 32MB RAM

For the Remote Administrative Console, these are the requirements:

- **FireWALL for NetWare and Windows NT:** If NDS integrated, the requirement is Windows 3.x, Windows95 or Windows NT 3.51 or later.
- **FireWALL for Windows NT:** Also manageable locally without a remote console.

## **Secure Computing's Sidewinder Firewall: a Type Enforcement Security**

Headquartered in St. Paul, Minn., Secure Computing is one of the largest network security companies in the world. Secure Computing's services and comprehensive suite of interoperable products address every aspect of enterprise network security including consulting services, firewalls, Internet monitoring and filtering, identification, authentication, authorization, accounting and encryption technologies. The only network security company that provides end-to-end network solutions

encompassing all universal enterprise security standards, Secure Computing has more than 4,000 customers worldwide, ranging from small businesses to Fortune 500 companies and government agencies.

Figure 14.131 is a screenshot of Secure Computing's Web site.

**Note:**

For more information, contact Secure Computing at 2675 Long Lake Road, Roseville, MN 55113. Tel +1.612.628.2700 Fax +1.612.628.2701. Or via e-mail at [sales@securecomputing.com](mailto:sales@securecomputing.com) or via the Web at <http://www.securecomputing.com>.

## ***The Sidewinder Security Server***

The Sidewinder Security Server is a network security gateway that stands between your internal computer network and the Internet and protects your network from unauthorized access. The Sidewinder uses Secure Computing's patented Type Enforcement security to ensure that attackers cannot infiltrate your protected network. For the past several years, the Sidewinder has been setting the industry standard in perimeter security.

The Sidewinder software runs on a Pentium-based computer with separate connections to a trusted and an untrusted network. Because it runs on standard hardware platforms and uses standard network interfaces, the Sidewinder can be integrated into almost any network configuration, as shown on figure 14.132.

The Sidewinder can give your organization the flexibility to implement and enforce even the most complex security policies. Sophisticated access controls and advanced filtering mechanisms allow you to control exactly who can access services through the firewall and what types of information they can transmit and receive. Encryption and authentication options provide even tighter security and allow organizations to create a virtual private network across the Internet.

An easy-to-use interface provides you, as an administrator, with a variety of tools for configuring and managing the Sidewinder, and the system can be administered locally or remotely. You can monitor network activity to detect unusual events that might indicate someone is trying to circumvent the security measures. You can also direct the Sidewinder to automatically gather information on these attempts and to try to identify the intruder.

By providing advanced technologies and filtering, the Sidewinder goes beyond traditional firewalls. It allows your organization to safely connect to an untrusted network and provides a gateway to help maximize an organization's Internet productivity.

## ***The Patented Type Enforcement Security***

Secure Computing's patented Type Enforcement technology, a key component of the Sidewinder Security Server, provides network security protection that is unique to the industry. Type Enforcement is software that greatly tightens security in the BSD UNIX operating system (BSD/OS) kernel, which is used on the Sidewinder. Implementing Type Enforcement within the operating system itself assures the highest level of security. It is impossible for any program executing on a Sidewinder with Type Enforcement to bypass the security features it provides.

The Sidewinder runs two different UNIX kernels that are used for different purposes. When the system is running and connected to its networks, it uses the operational kernel. When the operational kernel is booted, the Type Enforcement controls described in this section are in effect and cannot be disabled by any program running on the system. When an administrator needs to perform special tasks, such as restoring files, the Sidewinder runs in the administrative kernel. When the administrative kernel is running, the Sidewinder's network connections are disabled, so the system is isolated and protected.

The operational kernel divides the entire Sidewinder system into process domains and file types, as shown on figure 14.133.

Process domains are execution environments for applications such as FTP and Telnet. A process domain is set up to handle one kind of application, and each application runs in its own domain. File types are named groups of files and subdirectories.

A type can include any number of files, but each file on the system belongs to only one type.

Type Enforcement is based on the security principle of least privilege: any program executing on the system is given only the resources and privileges it needs to accomplish its task. On the Sidewinder, Type Enforcement enforces the least privilege concept by controlling the interactions between domains and file types, where,

- Each process domain on the Sidewinder is given access to only specific file types. If a process attempts to reference a file belonging to a type that it does not have explicit permission to access, the reference fails as though the file does not exist.
- Applications must usually collaborate with applications in other domains in order to do their job. On a typical system, this collaboration is done using the system's interprocess communications facility, which also opens up opportunities for breaching security. Type Enforcement eliminates this security risk by strictly controlling any communication between process domains. If a program in the process domain attempts to signal, or otherwise communicate with, a domain it does not have explicit permission to access, the communication attempt will fail.
- Most applications need to call operating system functions at times, but this can enable malicious users to access the kernel directly and compromise the system. To prevent this, Type Enforcement explicitly specifies which system functions can be called from each domain.
- One of the greatest security risks on a typical UNIX system is system administration, because of the high level of privileges needed to successfully manage and configure system resources. UNIX allows a user to log in as "super-user" (root), which gives the user access to all files and applications on the system. Under Type Enforcement, there is no super-user status. Each process domain is administered separately and is assigned its own administrative role. Each role is assigned only the privileges needed to administer a specific process domain. For example, if a user logs in using an account that is assigned the Web administrator role, that user cannot perform administrative tasks for mail or FTP.

Figure 14.134 illustrates how Type Enforcement controls a domain's access to files of different types. Any time a process tries to access a file, the Type Enforcement controls determine whether the access should be granted; these controls cannot be circumvented. In Figure 14.134, for example, a process running in Domain A is attempting to access File Type X; Type Enforcement denies this request. A process in domain B is permitted access to File Type X and File Type Z, while the process in domain C is granted access to File Type Y.

You can see the effects of Type Enforcement by looking at an example, such as mail services (mail services are notorious for security risks). Type Enforcement controls the mail server process by:

- Providing the mail process with access to only those files it needs to save and obtain mail.
- Permitting the mail process to communicate only with those processes it needs to transfer mail.
- Allowing the mail process to make only the system calls that are necessary for mail handling.
- Restricting mail administration capabilities to only those accounts that have been assigned the mail administration role.

Using the mail example, you can see how Type Enforcement provides restriction and containment. Even if an attacker managed to discover and exploit a weakness in the mail server, the attacker is restricted from entering another domain. Any resulting damage is contained within the mail domain, and applications executing in other domains are not affected. There is no way to gain access to the root directory, for example, or to break into any other part of the system.

## ***Remote Management***

The Sidewinder's remote management capability is crucial for solving the network administration concerns of large organizations with remote or branch offices. The ability to configure remote systems from a centralized location provides an additional layer of information security control. By adding strong authentication and virtual private network (VPN) capabilities to a Sidewinder, secure remote management becomes a reality.

## ***Access Controls***

The Sidewinder provides all of the basic Internet services your site needs, along with sophisticated controls that allow your organization to easily allow or deny user access to these services.

These controls are configured in the Access Control List (ACL), a database of configurable rules. Each rule determines whether or not a user program may open a connection to a network service proxy or a server application on the Sidewinder. The connection request may originate from either an internal network or the Internet. When a network connection is requested, the Sidewinder checks the ACL entries to determine whether to allow or deny the connection.

For example, your organization may want to allow all internal users to access the World Wide Web at any time, or you might want to allow Web access by only specific users on certain internal systems at certain times of the day. You may want to allow Internet users to access an FTP server located on the Sidewinder, or you may want to allow certain Internet users to access an internal system situated behind the Sidewinder.

The Sidewinder's interface provides an easy way of configuring ACL entries, as shown below. When the Sidewinder is installed, the initial ACL database contains entries that allow certain connections from the internal network to the Internet. You can then add, modify, or delete individual Access Control List entries and configure them as necessary according to the requirements of your organization's security policy. At any time, as shown on figure 14.135 you can quickly change the ACL entries to make new services available and to loosen or tighten access restrictions based on your organization's unique needs.

The ACL is extremely flexible and allows organizations to restrict connections based on the following criteria:

- **Source or destination burb** - A burb is a type-enforced network area used to isolate network interfaces from each other. You can allow or deny connections based on the source burb, the destination burb, or both.
- **Source or destination network object type or group** - You can allow or deny connections based on a source network object, a destination network object, or both, as shown on figure 14.136. A source or destination object can be an IP address, a host name, a domain name or a subnet. In addition, you can set up network groups composed of any combination of these objects. For example, you may want to allow Telnet access from several specific host computers and IP addresses residing on your internal network. You can easily create a group comprising these host names and IP addresses. Then, you can quickly create an ACL entry allowing Telnet access for this group rather than creating separate ACL entries for each host name and IP address.
- **Type of connection agent** - You can configure an ACL entry to allow or deny connections based on the software agent in the Sidewinder that is providing the connection. One type of agent is a proxy, which allows communication through the Sidewinder without any direct contact between systems on opposite sides of the firewall. A second type of agent is a server, which provides a service on the Sidewinder itself, such as FTP. The third type of agent is a Network Access Server (NAS), which provides dial-up connectivity from a bank of modems.
- **Type of requested network service** - You can allow or deny connections based on the type of service that is being requested. The Sidewinder provides proxies for most popular Internet services. These are pre-configured and set up to use standard port numbers. These include AOL, FTP, Web (http), Real Audio and Telnet. In addition, you can set up your own UDP or TCP proxy by configuring a port for a specific service. For example, you can set up a UDP proxy to allow you to route Simple Network Management Protocol (SNMP) messages through the Sidewinder.

You can also set up rules that are unique to some network services. For example, FTP can be controlled by a rule that allows only GET operations, thus preventing it from writing to the server. Similarly, you can control access to Web services based on a Web site's content using Secure Computing's SmartFilter™ technology.

- **User requesting the connection** - For services that support authentication (such as Web and FTP), you can restrict access based on the user requesting the connection. Authentication

You can set up a rule requiring the Sidewinder to authenticate the requester's identity before granting the connection request. You can use standard password authentication, or you can implement strong authentication to provide tighter security. Strong authentication methods that are supported include LOCKout DES, LOCKout FORTEZZA and the SafeWord Authentication Server, which are all premium features available for the Sidewinder. (See the "Premium Features" section for more information.) You can also use strong authentication provided by a Defender Security Server or an ACE/Server.

- **Time and day of the connection request** - You can specify the day and/or time of day when a connection is permitted. For example, you could allow internal access to certain Internet services during the times when your site's network traffic is lightest.
- **Encryption** - You can configure an ACL entry that requires the incoming connection request to be encrypted. This is a premium feature available when you purchase the Sidewinder's IPSEC software option. See the "Premium Features"

section for more information.

- **Redirection** - For added security on external-to-internal connections, you can redirect a connection. Setting up an ACL entry for redirection tells the Sidewinder to route the requested connection to a different address or a different port on the internal network. For example, if you wanted to allow external access to an internal FTP system, you would publish the Sidewinder's address as the address of that internal system. When someone attempts the connection, the Sidewinder would route it to the appropriate internal destination.

## ***Extensive Event Monitoring***

Event monitoring is the process of auditing security-related events and responding to them. Event monitoring is one of the most important Sidewinder features, because it provides you with both the means to detect possible intruders and the information you need to respond to the intrusions.

In Sidewinder terminology, an event is an abnormal security-related incident. For example, as shown on figure 14.137, a connection request from the Internet to an address on the internal network is abnormal, because the Sidewinder does not allow internal network addresses to be made public. An event such as this may mean that someone is probing in an attempt to gain access to the internal network.

The Sidewinder monitors seven different types of events:

- service denials
- attack attempts
- authentication failures for Telnet or FTP proxies
- mail messages that are rejected by a mail filter
- attempted network probes
- exceeded network traffic threshold
- attempts to circumvent Type Enforcement

When the Sidewinder detects one of these events, it responds based on controls set by the administrator. Since most events are unintentional, it isn't practical to respond to every one. When a particular event is repeated during a short time interval, however, it may indicate malicious intent that warrants action.

The Sidewinder administrator specifies when an event will trigger an alarm and when it will be ignored by setting up thresholds. For example, the administrator might specify that five network probe attempts in one hour will trigger an alarm.

## ***Advanced Filtering***

Even after authenticating users and restricting access to network resources, an enterprise's security may still be in jeopardy if unauthorized content is allowed to pass between connections. The Sidewinder provides what most security systems do not: advanced filtering technology that lets an organization prevent undesirable messages from flowing between networks.

The Sidewinder contains filtering mechanisms for three major areas of vulnerability:

- electronic mail
- Web pages
- Java applets

### **Email filtering**

An enterprise's email system can be critical to its success. On the other hand, there can be disastrous consequences if an organization's mail system is misused. To further secure the mail system, the Sidewinder provides three kinds of mail filters:

- A binary filter blocks mail that contains binary data such as MIME (multipurpose Internet mail extension) attachments.
- A key word filter blocks mail containing words the administrator specifies.



- A size filter blocks mail messages that are too long.

Using the Sidewinder's interface, an administrator can set up each mail filter individually. A mail map like the one shown below specifies the mail filters to use, the order of filtration and the actions that should be taken when a message passes or fails the filter test. The interface provides a handy tool for visually configuring mail maps, as shown on figure 14.138.

## Web page filtering

In addition to being a valuable source of information, Web pages are becoming increasingly important for Internet commerce and remote education. Indiscriminate access to Web pages, however, can lower productivity or even cause legal problems in an organization. The Sidewinder's SmartFilter™ technology allows your organization to capitalize on the Web's benefits while controlling Web access.

SmartFilter allows a Sidewinder administrator to specify which Web pages from the list provided should be inaccessible, as seen on figure 14.139. For example, you might choose to block access to Web sites dealing with crime, games or gambling. Note that you can combine this filtering with access control options to narrowly define when specific types of Web sites can be accessed. You might allow game sites to be available on weekends or during evenings, for instance.

## Java applet filtering

Java applets are essentially executable programs referenced by Web pages. While they are extremely useful for expanding the Web page capabilities, they can also be put to malicious uses, such as tying up a client machine's resources or duping a user into providing authentication passwords. To combat this threat, the Sidewinder allows you to deny download requests for Java files

# IBM's Internet Connection Secure Server Firewall: a Type Enforcement Security

Who doesn't know IBM? IBM creates, develops and manufactures the industry's most advanced information technologies, including computer systems, software, networking systems, storage devices and microelectronics.

IBM has two fundamental missions: they strive to lead in the creation, development and manufacture of the most advanced information technologies, and they translate advanced technologies into value for customers as the world's largest information services company. Their professionals worldwide provide expertise within specific industries, consulting services, systems integration and solution development and technical support..

Figure 14.140 is a screenshot of IBM's Web site.

### Note:

For more information, contact IBM North America, 1133 Westchester Avenue, White Plains NY 10604, telephone (520) 574 4600 or toll free number (for use within the United States) 1 800 IBM 3333. You can also contact them via e-mail at [ibm\\_direct@vnet.ibm.com](mailto:ibm_direct@vnet.ibm.com) or visit their Web site at <http://www.ics.raleigh.ibm.com/>

## The IBM Firewall V3.1 for AIX

IBM Firewall Version 3 Release 1.1 for AIX is the latest release of IBM's award-winning firewall. It is available for RS/6000 machines running AIX. This product reflects IBM's commitment to delivering versatile security solutions, implementing not just one of the firewall technologies but several. The flexibility of the IBM Firewall, the addition of an innovative graphical user interface, and powerful administration and management tools make the IBM Firewall a leader in Internet security offerings.

For over a decade, IBM has used the IBM Firewall to protect its own corporate networks. With access to the Internet from internal IBM networks, IBM can be confident that with the IBM Firewall in place those internal, secure networks will stay that way.

The IBM Firewall stops network intruders in their tracks. It combines all three leading firewall architectures (application proxies, SOCKS circuit gateway, and filtering) in one flexible, powerful security system. It runs on an IBM RS/6000 workstation with AIX Version 4.1.5 or 4.2. And, as an e-business enhancer, it supports the IBM Network Computing Framework for e-business.

The Java-based graphical user interface (GUI) offers an easy-to-use and safe tool for administrators. Easy-to-use because the interface is interactive and dynamic. Safe because Java applets are installed on the administrator's workstation instead of on the network.

Navigation through the interface itself is easy, thanks to a navigation tree that is always visible for guidance. Through this navigation tree, administrators can easily find their way around the GUI and move from one task to another. Help with using the GUI is available in several different forms, from context-sensitive help to immediate access to the online documentation. Figure 14.141 shows the main panel of the GUI.

The IBM Firewall also eases your administrative tasks. The Enterprise Firewall Manager allows several firewalls to be administered from a central location. And with administrators authorized for only specific tasks, you can maintain control over who does what.

## **Great Level of Protection**

The 56-bit data encryption standard (DES) is one of the strongest encryption techniques on the market. With federal government approval, the IBM Firewall is a leader in exporting this 56-bit key, enhancing the security of your networks and data.

In addition, the IBM Firewall offers a tool for scanning your networks, servers, and firewalls, looking for potential security gaps. This advanced tool, called Network Security Auditor, is a proactive means of maintaining a vigilant eye on your system.

## **Greater Accessibility**

Virtual private networks (VPNs) provide secure communication across the Internet. You can give remote users the same accessibility to internal networks while protecting their communication across the Internet. Client-to-firewall VPNs allow remote users to have private and secure communication even when the traffic travels over the Internet. These users can change ISP-assigned IP addresses without losing access.

The IBM Firewall uses state-of-the-art technology to deliver a flexible and versatile firewall solution, with application gateways, a Socks server, and advanced filtering capabilities. In one product, you have the choice of firewall technologies that best suit your needs. These technologies, combined with an innovative graphical user interface and powerful administration and management tools, make the IBM Firewall a leader in Internet security offerings.

## **IBM Firewall Filtering**

Filters are one way the IBM Firewall controls traffic from one network to another. The filters operate on criteria such as IP source or destination address range, TCP ports, UDP responses, Internet Control Message Protocol (ICMP) responses, and TCP responses.

## **IBM Firewall as an Application-Level Proxy**

The IBM Firewall application-level proxy is referred to as the proxy server. If a proxy server does not prompt a user for a password or other authentication, it is considered transparent. The IBM Firewall implements full proxy servers for Telnet and FTP as well as transparent proxy servers for Telnet, FTP, and HTTP.

A full proxy server is a secure server that runs on the firewall and performs a specific TCP/IP function on behalf of a network

user. The user contacts the proxy server using one of the TCP/IP applications (Telnet or FTP). The proxy server makes contact with that remote host on behalf of the user, thus controlling access while hiding your network structure from external users. Figure 14.142 illustrates a proxy Telnet server intercepting a request from an external user.

The IBM Firewall FTP and Telnet proxy servers can authenticate users with a variety of authentication methods, including password verification, SecurID cards, S/Key, and SecureNet Key cards.

## IBM Firewall as a Circuit-Level Proxy

The IBM Firewall implements circuit-level proxies in two ways: as a Socks server and through network address translation (NAT).

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 workstations, 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. Figure 14.143 illustrates the Socks server intercepting an HTTP request from a client within the secure network.

The other implementation of circuit-level proxy is network address translation (NAT) which can be used for both TCP- and UDP-based applications. With the explosive growth of the Internet, IP address depletion becomes a problem. NAT provides a solution.

The IBM Firewall manages a pool of IP addresses that can be used to communicate on the Internet. NAT translates secure IP addresses to temporary, external registered IP address from the address pool. This allows trusted networks with privately assigned IP addresses to have access to the Internet. This also means that you don't have to get a registered IP address for every machine in your network.

Both the Socks server and NAT effectively hide your internal IP addresses from the outside world.

## Use of Encryption

The IBM Firewall provides secure communication across a public network like the Internet through virtual private networks (VPNs). A VPN is a group of one or more secure IP tunnels. When two secure networks (each protected by a firewall) establish a VPN between them, the firewalls at each end encrypt and authenticate the traffic that passes between them. Likewise, when a VPN is established between a remote client and a firewall, the traffic between them is encrypted and authenticated. The exchange of data is controlled, secure, and validated.

## Managing the IBM Firewall

Implementing these firewall techniques helps you establish a perimeter defense around your network. You also need to monitor this defense and analyze events that take place at the firewall, watching for suspicious activity.

The IBM Firewall has sophisticated management capabilities that make creating and distributing your security policies through your organization secure yet simple. Key features for the administrator include a Java-based graphical user interface (GUI), the ability to manage multiple firewalls from a central location, the ability to assign different levels of authority so that administrators are authorized to do specific activities, and a tool that scans your firewall configuration looking for potential security exposures.

The IBM Firewall also provides logging, alerting, monitoring, and reporting facilities. For example, tools can monitor unauthorized attempts to access your system and perform an action you have defined when a certain threshold is reached (such as paging an administrator if more than five unauthorized attempts are recorded within a certain time limit). The reporting facilities build tables for a relational database tool, allowing you to generate reports.

# Main IBM Firewall Features

The IBM Firewall features can be grouped into these categories:

- Using firewall technology and security features
- Communicating through virtual private networks
- Using the Network Security Auditor
- Administering the firewall
- Logging, monitoring, alerting, and reporting
- Ensuring availability of the firewall

## Network Address Translation

Network address translation (NAT) solves the problem of Internet IP address depletion by allowing addresses inside your local IP network to be shared across your network.

When a user sends information to the Internet, the request goes to the firewall first. The firewall changes the internal IP address to a registered external IP address before the information goes out. When information comes back addressed to that external IP address, the IBM Firewall translates it back to the corresponding internal address. This translation process is shown in Figure 14.144.

Hiding your internal IP addresses from the outside world helps you in a few ways. It's tougher for hackers to get to your internal network because the structure of your internal network is hidden. For example, you might set up a numbering convention for IP addresses within your company. You don't have to worry about a competitor figuring out the convention and knowing more about your company than you want to reveal. Using NAT also keeps you from having to obtain registered IP addresses for every machine in your network, which would be extremely time consuming and costly.

NAT supports both UDP- and TCP-based applications.

## SafeMail

SafeMail is an IBM mail gateway. The SafeMail function does not store mail on the gateway or run under the root user ID. The firewall gateway name is substituted for the user's name on outgoing mail so that mail appears to be coming from the firewall's address instead of the user's address. SafeMail supports Simple Mail Transfer Protocol (SMTP) and Multipurpose Internet Mail Extensions (MIME).

## Strong Authentication

The IBM Firewall lets you choose from many methods for authenticating users. You can use just a password, but in certain situations this may not be secure enough. Particularly when logging in from the non-secure network, a password could easily be intercepted by a would-be intruder. The IBM Firewall provides a strong authentication method, Security Dynamics SecurID\*\* card, plus the opportunity to implement your own unique authentication method.

The method from Security Dynamics includes a user ID and a SecurID card. When you're logging in remotely, you get your password from the SecurID card. The password changes every 60 seconds and is good for one-time use only. So, even if someone does intercept your password over the open network, the password is not valid by the time the hacker gets it.

You can also customize a user exit to support any other authentication mechanism. The IBM Firewall includes an application programming interface (API) to help you define your own authentication technique.

## Hardening

When you install the IBM Firewall, there are some non-secure services and protocols embedded within UNIX and TCP/IP, along with accounts that could create a hole in your security policy. The IBM Firewall installation process disables these applications and non-secure UNIX accounts on the firewall machine. (This process is also known as hardening your

operating system.)

Once you have completed the installation and configuration, a background program periodically checks for altered configuration files. A message is sent to the syslog and an alarm is generated when this program detects that the protected files were changed.

## ***Communicating through Virtual Private Networks***

Suppose you want to use the Internet instead of leased lines to communicate with your suppliers or business partners who don't have direct access into your corporate network. The IBM Firewall virtual private network offers you protection against eavesdropping.

A virtual private network (VPN) is a group of one or more secure IP tunnels. A secure IP tunnel permits a private communication channel between two private networks over a public network such as the Internet. The two private networks are each protected by a firewall. The two firewall machines establish a connection between them. They encrypt and authenticate traffic passing between the private networks. The IBM Firewall follows IPSec standards, and therefore is interoperable with other firewalls. Figure 14.145 shows a client-to-firewall tunnel as well as firewall-to-firewall tunnels.

## **Using the Network Security Auditor**

The Network Security Auditor scans your network for security holes or configuration errors. The Network Security Auditor scans your servers and firewalls for a list of problems or vulnerabilities, such as open ports and other exposures, and compiles a list so you can make corrections. The Network Security Auditor can be used as a periodic scanner of critical hosts or as a one-time information gathering tool. Administration of the Network Security Auditor is done through an easy-to-use HTML interface. With the Network Security Auditor, you maintain vigilance over your firewall.

Features of the Network Security Auditor include:

- Scanning TCP and UDP ports
- Recognizing servers on non-standard ports
- Reporting dangerous services, known vulnerabilities, obsolete server versions, and servers or services in violation of customized site policy
- Generating reports in HTML for easy browsing

Figure 14.146 shows Network Security Auditor sample output.

## ***Administering the Firewall***

The IBM Firewall presents a Java- and HTML-based graphical user interface (GUI) to administer a firewall. You can administer the firewall from Netscape 3.0 for AIX, which is included in the IBM Firewall package.

The GUI is easy for the firewall administrator to use. A navigation tree always appears on the left side so you can move around the GUI and easily go from one task to another. Figure 14.147 shows the main panel of the GUI.

## **Enterprise Firewall Manager**

The Enterprise Firewall Manager (EFM) allows you to administer multiple firewalls from one place. You can administer each firewall individually, or you can designate one firewall to be the central server to maintain the configuration files for all the firewalls. You can clone firewalls to create new ones, and you can replace configuration files with updated files whenever needed. In Figure 14.148, EFM is used to administer two firewalls (A and B) that are within the same secure network as the EFM and one remote firewall (C) that is in a different secure network.

# System requirements

The following is a list of system requirements to run IBM the Firewall:

- A RISC System/6000
- At least two communication adapters, supported by the TCP/IP protocol stack
- 64MB of memory
- 800-1000MB of disk space
- AIX, Version 4.1.5 or 4.2

Orders

Orders

Back

Forward

Comments

Comments

---

[COMPUTING MCGRAW-HILL](#) | [Beta Books](#) | [Contact Us](#) | [Order Information](#) | [Online Catalog](#)

---

HTML conversions by [Mega Space](#).

This page updated on December 05, 1997 by [Webmaster](#).

[Computing McGraw-Hill](#) is an imprint of the [McGraw-Hill Professional Book Group](#).

Copyright ©1997 [The McGraw-Hill Companies, Inc.](#) All Rights Reserved.

Any use is subject to the rules stated in the [Terms of Use](#).

[Orders](#)[Orders](#)[Back](#)[Comments](#)[Comments](#)

© 1997 The McGraw-Hill Companies, Inc. All rights reserved.

Any use of this Beta Book is subject to the rules stated in the [Terms of Use](#).

---

## Appendix A:

# List of Firewall Resellers and Related Tools

The following is a list of the main companies providing sales, VAR, and consulting on firewalls on Web sites and networks. There are companies with expertise on different operation systems and environment. The list is in alphabetical order. The technical information was provided by the company developing the product, extracted as courtesy of Catherine Fulmer from the URL:

<http://www.access.digex.net/~bdboyle/firewall.vendor.html>.

## AlterNet:

AlterNet is now offering security consulting services.

Bob Stratton Voice +1 703 204 8000

UUNET Technologies, Inc.

Email: [strat@uunet.uu.net](mailto:strat@uunet.uu.net)

## Atlantic Computing Technology Corporation

Atlantic is an authorized reseller for the BorderWare firewall server. Based out of Connecticut, Atlantic specializes in providing turn-key Internet security solutions -- helping to integrate existing network infrastructure with a secure firewall implementation.

Atlantic Computing Technology Corporation

84 Round Hill Road

Wethersfield, CT 06109

(203) 257 7163

Email: [info@atlantic.com](mailto:info@atlantic.com)

<http://www.atlantic.com>

## **ARTICON Information Systems GmbH**

ARTICON Information Systems is a reseller of the BorderWare Firewall for Web sites, also provides firewall design services.

<http://www.bell-atl.com>

## **Cisco Routers**

<http://www.cisco.com>

## **Cohesive Systems**

Cohesive Systems provides many networking and security services, as well as is a reseller of Trusted Information Systems Gauntlet.

Cohesive Systems is a leading network consulting firm that puts all the pieces together for corporate internetworks.

We partner with our clients to help them find technology solutions for their operating and business goals. We do this by providing the expertise products and services to build high performance information systems. Although we have expertise in all areas of network computing, we are widely recognized in the industry for Internet connectivity and security solutions.

Cohesive Systems Branden L. Spikes

1510 Fashion Island Blvd, Webmaster

Suite 104, San Mateo, CA 415-574-3500

Email: [bspikes@cohesive.com](mailto:bspikes@cohesive.com) - [info@cohesive.com](mailto:info@cohesive.com)

<http://www.cohesive.com>

## **Collage Communications, Inc.**

Collage Communications, Inc. is one of a few Non-Partisan suppliers for Internetworking Solutions. We handle six different firewall platforms, all the interconnect hardware including the Routers, Hubs, and Channel Interfaces. This allows us to react to a customers needs much more rapidly and fit a solution not only within their budget but tailored to their specific needs.

With regard to the CyberGuard platform we work with HCSC and represent them on this product locally



in the Northern California Territory.

Email: [cyberguard@CollCom.COM](mailto:cyberguard@CollCom.COM)

Collage COmmunications, Inc.

12 Tulip Lane

Palo Alto, Ca., 94303

## Conjungi Corporation

Conjungi Corporation is a re-seller for Trusted Information System's Gauntlet product. Located in Seattle, Washington, and doing business through the US and (increasingly) internationally.

Email: [simon@conjungi.com](mailto:simon@conjungi.com)

<http://www.conjungi.com>

## Cypress Systems Corporation, (Raptor reseller)

P. O. Box 9070

McLean, VA 22102-0070

Phone: (703) 273-2150

FAX: (703) 273-2151

e-mail: [rmck@sandfiddler.paragon-systems.com](mailto:rmck@sandfiddler.paragon-systems.com)

## Data General Corp. (Gauntlet Reseller)

Data General Corporation offers the Gauntlet Firewall Package.

Integrating software from Trusted Information Systems, Inc. with hardware, installation and support provided by Data General. Data General is an open systems company which specializes in providing servers, storage products and services to information systems users worldwide.

Gauntlet is a trademark of Trusted Information Systems, Inc.

Data General Corporation

4400 Computer Drive

Westboro, MA 01580

Tel: (800) 4DG-OPEN

<http://www.dg.com>

email: [sense@dg.com](mailto:sense@dg.com)

## **Decision-Science Applications, Inc.**

Decision-Science Applications, Inc. is a reseller of the BorderWare Firewall Server.

Decision-Science Applications, Inc.

1110 N. Glebe Rd., Suite 400

Arlington, VA 22201

Voice: (703) 875-9206

FAX: (703) 875-9585

Email: [infosec@dsava.com](mailto:infosec@dsava.com)

## **E92 PLUS LTD**

E92 PLUS LTD

St. James House

9-15 St.James Road

Surbiton, Surrey KT6 4QH

United Kingdom

tel: +44 (0) 181 399 3111

fax: +44 (0) 181 399 5111

Email: [e92plus@e92plus.co.uk](mailto:e92plus@e92plus.co.uk)

## **Enterprise System Solutions, Inc.(BorderWare reseller)**

<http://www.essi.com>

## **E.S.N - Serviço e Comércio de Informática Ltda.**

E.S.N. - Serviço e Comércio de Informática Ltda.

Rua Senador Dantas 117 Sala 1412

Centro

Rio De Janeiro - RJ

Brasil

tel: +55 21 262 1168

## FSA Corporation

FSA Corporation is a software company that is dedicated to providing security software for heterogeneous UNIX networks and PCs.

<http://www.fsa.ca>

## IConNet

IConNet is a full service Internet provider in NYC. We sell IP service, consulting, hardware and software dealing specifically with the Internet.

1. **Internet in a Rack (IR)** - IR is a full-service solution for corporate access, which includes a netra server, a dedicated Sparc 5 firewall (Checkpoint/SunSoft), a Cisco router, and a T1 CSU/DSU. It allows companies to connect to the Internet securely and quickly - setting up the system involves plugging in 4 cables and flipping a switch.
2. Netra servers, Checkpoint Firewall-1 software, and other security products and services. We are also a VAR for Cisco, chipcom, Sun, and many other high-end vendors.

Email: [info@iconnet.net](mailto:info@iconnet.net)

## Igateway by Sun Consulting.

Actually called CONSULT-IGATEWAY and consists of telnet and ftp proxies for filtered traffic. Available through Sun Consulting only.

## Ingress Consulting Group, LTD

BorderWare/JANUS Reseller Ingress Consulting Group LTD (BorderWare/Janus Reseller)

Empire State Building

Suite 3406

New York, NY 10018

800-254-7159 (voice)

508-349-0132 (fax)

New England Office

240 Zoheth Smith Way

Wellfleet, MA 02667

email: [ter@ingress.com](mailto:ter@ingress.com)

<http://www.ingress.com>

## INTERNET GmbH

German distributor of the BorderWare Firewall-Server and also a provider of consulting for firewalls, ISDN, Web Server, etc...

INTERNET GmbH

Am Burgacker 23 phone : +49-6201-3999-59

D-69488 Birkenau fax : +49-6201-3999-99

Germany

Contact: Ingmar Schraub

Web: GmbH

## Jeff Flynn & Associates

Jeff Flynn & Associates is a Network Consulting Service specializing in the design and implementation of Secure Networks. To remain unbiased, we do not resell products. On request, however, we will manage the selection, procurement, and installation of network security systems (e.g., firewalls, authentication, encryption, physical security, employee awareness programs, policies and procedures, etc.).

Phone: (714)551-6398

Jeff Flynn & Associates

19 Perryville

Irvine, Calif. 92720 USA

Email: [us028272@interramp.com](mailto:us028272@interramp.com)

## Media Communications eur ab, (Gauntlet Reseller)

We currently resell TIS's gauntlet and provide general computer security consulting.

Media Communications eur ab, box 1144, 111 81 stockholm, SWEDEN.

Neil Costigan

Email: [neil@medcom.se](mailto:neil@medcom.se)

<http://www.medcom.se>

ph: +46.708.432224 (GSM)

fax: +46.8.219505

video: +46.8.4402255 (h.320, isdn, up to 384k)

## **Mergent International, Inc. (Gauntlet Reseller)**

We are a comprehensive security solution vendor and leading provider of PC-based security solutions for distributed computing environments and an authorized reseller of the Gauntlet Internet firewall from Trusted Information Systems, TIS.

Mergent International, Inc., has the ability to provide the Gauntlet Internet firewall as a point software and/or hardware/software turnkey solution with the additional ability of integrating the firewall with Mergent's present suite of security products and services as a one-stop secure enterprise solution vendor.

For more information please visit us via:

<http://www.mergent.com>

Email: [info@mergent.com](mailto:info@mergent.com)

or call toll free 1.800.688.1199

## **Momentum Pty Ltd**

We are the leading open systems and Internet consulting organization in Western Australia. Our primary focus is Internet security and firewall systems. We supply a range of related services and software to corporate and government organizations.

For further information please contact us as follows:

Todd Hooper Marketing Director

Momentum Pty Ltd

PO Box 1436

Subiaco, WA 6904, Australia

Phone: +61 9 483 2649

Fax : +61 9 380 4371

EMail: [sales@momentum.com.au](mailto:sales@momentum.com.au)

<http://www.momentum.com.au>

## **NetPartners (Phil Trubey), (JANUS Reseller)**

Phone: 800-723-1166, 714-252-5493

Fax: 714-759-1644

E-Mail: [sales@netpart.com](mailto:sales@netpart.com)

## **Network Translation Services**

Our company, Network Translation, Inc., has such a Network Address Translation product (see RFC-1631). Give us a call, or check our web site:

<http://www.translation.com>

John Mayes

Network Translation, Inc.

415/494-NETS

## **OpenSystems, Inc.**

OpenSystems Inc. is a consulting and integration firm specializing in the design and deployment of network computing technologies for the corporate enterprise.

Our background and history as a provider of corporate computing solutions has helped us cultivate a unique set of skills in developing secure computing environments for customers with a critical need to protect corporate data.

This experience has enabled us to build expertise and develop a proven methodology and techniques in performing security assessments, policy reviews, designing enterprise security architectures, and rapidly deploying security solutions in the areas of computer, network, and Internet/Intranet security.

OpenSystems Inc. represents firewall products from Raptor Systems, Checkpoint/SunSoft and Sun Microsystems on UNIX and Windows NT. OpenSystems Inc. provides both bundled hardware/software/integration/training packages as well as custom consulting solutions.

OpenSystems Inc.

10210 NE Points Drive, Suite 110

Kirkland, WA 98033-7872

(206) 803-5000 / (206) 803-5001 FAX

E-Mail: [steve@opensys.com](mailto:steve@opensys.com)

<http://www.opensys.com>

## PDC

Peripheral Devices Corp.

<http://PDC>

Email: [mktg@pdc.com](mailto:mktg@pdc.com)

## PENTA

PENTA, Inc. Phone: (800) PENTA-79

333 North Sam Houston Parkway East (713) 999-0093

Suite 680 Fax: (713) 999-0094

Houston, TX 77060

## PRC

PRC is a leading integrator of open systems with over 40 years of experience in delivering quality results. We specialize in providing custom client/server solutions for your enterprise. Talk to us today about your Internet firewall requirements and be surprised at how easy it is to operate securely.

(Enterprise Assurance is a service mark of PRC).

Jay Heiser

Product Manager

Enterprise Assurance

1500 PRC Drive

McLean, Va 22102

703.556.2991

e-mail: [secure@prc.com](mailto:secure@prc.com)

<http://www.c3i.wsoc.com>

## Racal-Airtech Ltd, (Eagle reseller)

Racal is a world leader in information security and has a comprehensive portfolio of computer security products and services. Racals global security consultancy and support organizations together with its

world-leading combination of products and services, address the information security needs of financial institutions, government departments and commercial organizations - wherever they are located.

Racals integrated software and hardware security products each protect points of potential weakness - building into full end to end security solutions tailored to meet the information security needs of your organization.

As part of our commitment to provide "best of breed" products in a fast changing environment, Racal is pleased to be working with Raptor systems in supplying and supporting the EAGLE family of firewall software to our large customer base in the finance, government and commercial markets.

\* \* \* UK \* \* \*

Racal Airtech Ltd

Meadow View House

Long Crendon, Aylesbury

Buckinghamshire, United Kindom HP18 9EQ

Phone: 01844 201800

\* \* \* USA \* \* \*

Racal Gaurdata Inc

480 Spring Park Place

Suite 900

HERNDON, VA 22070

Phone: 703 471 0892

\* \* \*

Sohbat Ali

<http://www.gold.net>

## RealTech Systems

RealTech Systems Corporation is a systems integration company, located in NY city in the Empire State building and Albany NY, serving the needs of Fortune 500 companies.

RTS is CISCO Gold authorized, an Advanced Technical Partner of Bay Networks, a Platinum Novell reseller, and an authorized reseller of Checkpoints FireWall-1 product. Recent clients who RTS has completed Internet projects for include: Deloitte & Touche LLP, Hearst Magazines, and Standard Microsystems Corporation. Visit our web site at

<http://WWW.REALTECH.COM>



Email questions to [jhorowitz@realtech.com](mailto:jhorowitz@realtech.com)

Or call 212-695-7100 extension 2106

## Sea Change Corporation, (JANUS reseller)

6695 Millcreek Drive, Unit 8

Mississauga, Ontario, Canada L5N 5R8

Tel: 905-542-9484 Fax: 905-542-9479

Internet: [jalsop@seachange.com](mailto:jalsop@seachange.com)

<http://www.seachange.com>

\*\*\*\* Sea Change Corporation - Pacific Region

5159 Beckton Road 604-658-5448

Victoria, British Columbia V8y 2C2

Email: [michael@seawest.seachange.com](mailto:michael@seawest.seachange.com)

[peter@sea-europe.co.uk](mailto:peter@sea-europe.co.uk)

<http://www.sea-europe.co.uk>

## Security Dynamics Technologies

Security Dynamics Technologies Inc. (NASDAQ: SDTI), a leading provider of network and computer security solutions, today announced the activation of its Internet World Wide Web site:

<http://www.securid.com>

Offerings include product, partnership and general corporate information. Intending to be the leading Web site for information security content, future Security Dynamics' site offerings will include in-depth industry and technology analyses and customer-related support and services.

Site Name: Security Dynamics - SecurID

Organization: Security Dynamics Technologies, Inc.

Company Contact: Stew Guernsey

Address: One Alewife Center - 3rd Floor

Cambridge, MA 02140

Country: USA

Phone Number: (617) 234-7414

Stew Guernsey

## Softway Pty Ltd, (Gauntlet Reseller)

Softway is Australia's largest open systems consulting and services house. We are resellers for TIS' Gauntlet firewall and also provide a range of Unix and Network Security services.

Softway Pty Ltd Phone: +61 2 698 2322

P.O. Box 305 Fax: +61 2 699 9174

Strawberry Hills

NSW 2012

AUSTRALIA

Email: [enquiries@softway.com.au](mailto:enquiries@softway.com.au)

<http://www.softway.com.au>

## Spanning Tree Technologies Network Security Analysis Tool

Spanning Tree's NetProbe is an easy-to-install and use network security analysis tool that, from a single host, scans a network and tests the effective configuration for remote access vulnerabilities.

Designed to scan large networks with unequaled speed. NetProbe tests security from inside and outside firewalls. NetProbe reports and describes all discovered vulnerabilities and their fixes, with links to specific CERT/CIAC documents. Modular design allows fast updates from CERT/CIAC advisories. Reports can be tailored to your needs and all data is stored in a database for easy reference.

Encryption within the licensing mechanism prevents unauthorized use of NetProbe.

Spanning Tree Technologies, Inc. (515) 296-6900

2501 N. Loop Drive Fax: (515) 296-9910

Ames, IA 50010

Email: [info@spanning.com](mailto:info@spanning.com)

<http://www.spanning.com>

### *c. Stalker by Haystack Labs, Inc.*

This is an intrusion detection system. Stalker sets up, manages, reports, and analyzes audit trails from a variety of UNIX vendors. Stalker is used by the most paranoid organizations in several countries (including the US).

Stalker supports all Sun operating systems (SunOS, Solaris, and Sun Trusted Solaris) and IBM AIX. We have ports underway to additional platforms, including HP, and are working on several network and router monitoring tools, as well. And our customers benefit from receiving ongoing updates to our Misuse Detection Database.

For more information, email [info@haystack.com](mailto:info@haystack.com) or call us:

Haystack Labs, Inc.

10713 RR620N, Suite 521

Austin, TX 78726 USA

(512) 918-3555 (voice)

(512) 918-1265 (fax)

## Stonesoft Corporation

Stonesoft Corporation in Finland, a FW-1 reseller.

Taivalmäki 9 FIN-02200 Espoo, Finland

phone: +358 0 4767 11- fax: +358 0 4767 1234

phone: +358 0 422 400 - fax: +358 0 422 110

email: [info@stone.fi](mailto:info@stone.fi)

## TeleCommerce

TeleCommerce is a Network Systems Corp. VAR in S. California. We specialize in Virtual Private Networks over the Internet to replace costly dedicated and leased lines.

Email: [info@TeleCommerce.com](mailto:info@TeleCommerce.com)

<http://WWW.TeleCommerce.com>

Phone: 805-289-0300

## Trident Data Systems, (SunScreen provider)

Trident is an authorized SunScreen service provider, offering security policy development, equipment installation, and system administration training to customers of Sun who have chosen to secure their critical communications links with the SPF100. We are also a member of the highly selective SunIntegration Alliance Program, providing multi-platform rightsizing and client-server migration to our clients.

Regional Sales Manager Dave Fuino

1330 Inverness Drive, Suite 310 (800)342-5831

Colorado Springs, CO 80910 Fax: (719)597-7234

EMail: Dave Fuino

Email: Steve\_McConnell@sablanco.tds.com

<http://www.tds.com>

***c. Tripcom Systems Inc.***

Reseller for CheckPoint's FireWall-1 product in the Chicago area. Also complete consulting and implementation services for Internet connectivity.

Tripcom Systems Inc.

Naperville, IL

708-778-9531

E-Mail: Adam Horwitz

## **Trusted Network Solutions (Pty) Ltd.**

Trusted Network Solutions (Pty) Ltd, based in Johannesburg, South Africa, focuses on all aspects of network security. This includes security audits, secure e-mail products, link encryption, network firewalls, and dial-in security solutions. TNS is a value-added reseller of the Gauntlet firewall. For more information -

<http://www.tns.co.za/tns>

## **UNIXPAC AUSTRALIA**

UNIXPAC, headquartered in Cremorne (Sydney), NSW, Australia markets, and services enterprise-wide systems solutions for internetworking, fire-wall security and data protection. Unixpac are the Australian agents for Raptor Systems (Eagle firewall) and ISS (Internet Scanner).

Unixpac can be contacted directly at (02) 9953 8366, toll free number 1 800 022 137, or via Internet e-mail at: [info@unixpac.com.au](mailto:info@unixpac.com.au)

<http://www.unixpac.com.au>

## **X + Open Systems Pty Ltd., (Internet Consultants)**

X + Open Systems is a highly specialized open systems consulting organization. Providing security/risk analysis, firewall design/construction plus a range of UNIX, networking and open systems related services.

X + Open Systems Pty Ltd Phone: +61 2 9957 6152

P.O. Box 6456 Shoppingworld Fax: +61 2 699 9174

North Sydney

NSW 2059

AUSTRALIA

Email: [info@xplus.com.AU](mailto:info@xplus.com.AU)

## **Zeuros Limited**

Zeuros Limited in Rotherwick, Hampshire, England have been supplying the Raptor Eagle Firewall for over 12 months into Banking, Telecommunications and other major UK corporates. Primarily a facility management company, the provision of secure data networking, Internet protection and secure virtual private networking services has fitted easily into Zeuros's portfolio.

For information, sales and support on the Raptor Eagle in the UK contact:

<http://www.zeuros.co.uk>

Les Carleton

Zeuros Limited

Tudor Barn, Frog Lane

Rotherwick, Hampshire, RG27 9BE

Tel:- 44 (0) 1256 760081

Fax:- 44 (0) 1256 760091

Email: [les@zeuros.co.uk](mailto:les@zeuros.co.uk)

## ***Firewall Tools: Public Domain and Shareware, Etc.***

## **Drawbridge**

Available at [net.tamu.edu](http://net.tamu.edu)

# Freestone by SOS Corporation

Freestone is a freely available application gateway firewall package. Freestone is a genetic derivative of Brimstone and was produced by SOS Corporation. Freestone can be retrieved from the Columbia, SOS, and COAST FTP sites.

## fwtk - TIS Firewall Toolkit

Available from ftp.tis.com. Look in /pub/firewalls and /pub/firewalls/toolkit for documentation and toolkit.

## ISS

Internet Security Scanner is an auditing package that is publicly available that checks domains and nodes searching for well-known vulnerabilities and generating a log for the administrator to take corrective measures. The publicly available version is on aql.gatech.edu /pub/security/iss.

## SOCKS

The SOCKS package, developed by David Koplas and Ying Da Lee. Available by ftp from ftp.nec.com.

|        |        |
|--------|--------|
| Orders | Orders |
|--------|--------|

Back

|          |          |
|----------|----------|
| Comments | Comments |
|----------|----------|

---

[COMPUTING MCGRAW-HILL](#) | [Beta Books](#) | [Contact Us](#) | [Order Information](#) | [Online Catalog](#)

---



HTML conversions by [Mega Space](#).

This page updated on December 05, 1997 by [Webmaster](#).

[Computing McGraw-Hill](#) is an imprint of the [McGraw-Hill Professional Book Group](#).

Copyright ©1997 [The McGraw-Hill Companies, Inc.](#) All Rights Reserved.

Any use is subject to the rules stated in the [Terms of Use](#).