

Contents

Using the GNU General Public License

The GNU General Public License (GPL) is the basis of the open source movement. This license is provided by the Gnu is Not Unix (GNU) organization, which develops various software packages. The most important element of this license is that instead of protecting a particular person or company, it protects the software code that creates the application.

Foreword

xxvii

Chapter 1 Introduction to Open Source Security

1

Introduction	2
The Tools Used in This Book	3
Using the GNU General Public License	3
Fee-Based GPL Software	5
Can I Use GPL Software in My Company?	5
Soft Skills: Coping with Open Source Quirks	6
General Lack of Installation and Configuration Support	6
Infrequent or Irregular Update Schedules	6
Command-Line Dominance	6
Lack of Backward Compatibility and No Regular Distribution Body	7
Inconvenient Upgrade Paths	7
Conflicts in Supporting Libraries and Limited Platform Support	7
Interface Changes	8
Partially Developed Solutions	8
Should I Use an RPM or Tarballs?	10
Tarball	10
Red Hat Package Manager	11
Debian	11
Obtaining Open Source Software	12
SourceForge	12
Freshmeat	13
Packetstorm	14

xi

SecurityFocus	15
Is That Download Safe?	16
A Brief Encryption Review	16
Symmetric Key Encryption	17
Asymmetric Key Encryption	18
Public Key and Trust Relationships	19
One-Way Encryption	20
GNU Privacy Guard	21
Deploying GNU Privacy Guard	21
Skipping Public Key Verification	29
Using GPG to Verify Signatures on	
Tarball Packages	30
Using Md5sum	30
Auditing Procedures	31
Locking Down Your Network Hosts	31
Securing Data across the Network	32
Protecting the Network Perimeter	33
Summary	35
Solutions Fast Track	35
Frequently Asked Questions	38

Chapter 2 Hardening the Operating System 41

Introduction	42
Updating the Operating System	42
Red Hat Linux Errata and Update Service	
Packages	42
Handling Maintenance Issues	43
Red Hat Linux Errata: Fixes and Advisories	44
Bug Fix Case Study	46
Manually Disabling Unnecessary Services	
and Ports	47
Services to Disable	47
The xinetd.conf File	48
Locking Down Ports	50
Well-Known and Registered Ports	50
Determining Ports to Block	52

Determining Which Ports to Block



When determining which ports to block on your server, you must first determine which services you require. In most cases, block all ports that are not exclusively required by these services. This is tricky, because you can easily block yourself from services you need, especially services that use ephemeral ports. If your server is an exclusive e-mail server running SMTP and IMAP, you can block all TCP ports except ports 25 and 143, respectively. If your server is an exclusive HTTP server, you can block all ports except TCP port 80.

Blocking Ports	53
Xinetd Services	53
Stand-Alone Services	54
Hardening the System with Bastille	55
Bastille Functions	55
Bastille Versions	63
Implementing Bastille	64
Undoing Bastille Changes	74
Controlling and Auditing Root Access with Sudo	77
System Requirements	79
The Sudo Command	79
Downloading Sudo	80
Installing Sudo	82
Configuring Sudo	86
Running Sudo	90
No Password	92
Sudo Logging	93
Managing Your Log Files	96
Using Logging Enhancers	97
SWATCH	97
Scanlogd	100
Syslogd-ng	101
Summary	103
Solutions Fast Track	104
Frequently Asked Questions	107

Chapter 3 System Scanning and Probing 109

Introduction	110
Scanning for Viruses Using the AntiVir Antiviru	
Application	110
Understanding Linux Viruses	110
Using AntiVir	112
Key Mode and Non-Key Mode	114
Licensing AntiVir	114
Exercise: Updating AntiVir	114
Using TkAntivir	116
Required Libraries and Settings	117

Learn How to Set Preferences For TkAntivir



Scanning Systems for Boot Sector and E-Mail Viruses	117
Additional Information	120
Exercise: Using TkAntivir	120
Scanning Systems for DDoS Attack Software	
Using a Zombie Zapper	123
How Zombies Work and How to Stop Them	124
When Should I Use a Zombie Zapper?	125
What Zombie Zapper Should I Use?	125
What Does Zombie Zapper Require to Compile?	127
Exercise: Using Zombie Zapper	127
Scanning System Ports Using the Gnome Service	
Scan Port Scanner	129
Required Libraries	130
Why Use a Port Scanner?	131
Exercise: Using Gnome Service Scanner	131
Using Nmap	133
Isn't Nmap Just Another Port Scanner?	134
Acquiring and Installing Nmap	136
Common Nmap Options	136
Applied Examples	137
Scanning Entire Networks and Subnets	138
Selective Scanning	139
Adding More Stealth	139
Saving to Text and Reading from Text	140
Testing Firewalls and Intrusion Detection Systems	141
Example: Spoofing the Source Address of a Scan	142
Timing Your Scan Speeds	142
Example: Conducting a Paranoid Scan	143
Exercise: Using Nmap	143
Using Nmap in Interactive Mode	144
Exercise: Using Nmap in Interactive Mode	144

Using NmapFE as a Graphical Front End	146
Exercise: Using NmapFE	147
Using Remote Nmap (Rnmap) as a Central Scanning Device	147
Exercise: Scanning Systems with Rnmap	148
Deploying Cheops to Monitor Your Network	151
How Cheops Works	153
Obtaining Cheops	154
Required Libraries	154
The Cheops Interface	155
Mapping Relations between Computers	157
Cheops Monitoring Methods	157
Connectivity Features	159
Exercise: Installing and Configuring Cheops	160
Deploying Nessus to Test Daemon Security	165
The Nessus Client/Server Relationship	167
Windows Nessus Clients	169
Required Libraries	169
Order of Installation	170
Configuring Plug-Ins	173
Creating a New Nessus User	174
The Rules Database	174
Exercise: Installing Nessus and Conducting a Vulnerability Scan	175
Updating Nessus	179
Understanding Differential, Detached, and Continuous Scans	180
Exercise: Conducting Detached and Differential Scans with Nessus	182
Summary	185
Solutions Fast Track	185
Frequently Asked Questions	189

Chapter 4 Implementing an Intrusion Detection System 191

Introduction	192
Understanding IDS Strategies and Types	194
IDS Types	195
Host-Based IDS Applications	196
Network-Based IDS Applications	196
IDS Applications and Fault Tolerance	197
What Can an IDS Do for Me?	200
Which IDS Strategy Is Best?	203
Network-Based IDS Applications and	
Firewalls	203
IDS Applications	204
Installing Tripwire to Detect File Changes on	
Your Operating System	206
Tripwire Dependencies	207
Availability	208
Deploying Tripwire	208
Tripwire Files	208
Tripwire Installation Steps	209
Configuring the Tripwire Policy File	209
Creating the Tripwire Policy File	212
Database Initialization Mode	212
Testing E-Mail Capability	214
Integrity Checking Mode	214
Specifying a Different Database	215
Reading Reports	215
Updating Tripwire to Account for Legitimate	
Changes in the OS	215
Updating the Policy	216
What Do I Do if I Find a Discrepancy?	217
Configuring Tripwire to Inform You Concerning	
Changes	217
Exercise: Installing Tripwire	217
Exercise: Securing the Tripwire Database	219
Exercise: Using Cron to Run Tripwire	
Automatically	220

SECURITY ALERT!

Although Tripwire has a "file integrity mode," Tripwire is not really an integrity checker in the classic sense. It does not, for example, test the file's stability or inode number or any other aspect in regards to file storage. Tripwire simply compares a file's new signature with that taken when the database was created. Other tools may be used to check the integrity of a file's permissions and ownership information.

Deploying PortSentry to Act as a	
Host-Based IDS	220
Important PortSentry Files	221
Installing PortSentry	222
Configuring PortSentry to Block Users	222
Optimizing PortSentry to Sense Attack Types	223
Exercise: Installing and Configuring	
PortSentry	224
Exercise: Clearing Ipchains Rules	227
Exercise: Running an External Command	
Using PortSentry	227
Installing and Configuring Snort	229
Availability	229
Supporting Libraries	229
Understanding Snort Rules	230
Snort Variables	230
Snort Files and Directories	231
Snort Plug-Ins	232
Starting Snort	233
Logging Snort Entries	236
Running Snort as a Network-Based IDS	236
Ignoring Hosts	237
Additional Logging Options: Text	
files, Tcpdump, and Databases	237
Configuring Snort to Log to a Database	238
Controlling Logging and Alerts	239
Getting Information	240
Exercise: Installing Snort	240
Exercise: Using Snort as an IDS	
Application	241
Exercise: Configuring Snort to Log to	
a Database	243
Exercise: Querying a Snort Database	
from a Remote Host	251
Identifying Snort Add-Ons	251
SnortSnarf	252

Exercise: Using SnortSnarf to Read	
Snort Logs	252
Analysis Console for Intrusion Databases	252
Summary	254
Solutions Fast Track	254
Frequently Asked Questions	258

Chapter 5 Troubleshooting the Network with Sniffers **261**

Introduction	262
Understanding Packet Analysis and TCP	
Handshakes	264
TCP Handshakes	265
Establishing a TCP Connection	265
Terminating a TCP Connection	266
Creating Filters Using Tcpdump	268
Tcpdump Options	268
Tcpdump Expressions	271
Boolean Operators	275
Installing and Using Tcpdump	276
Configuring Ethereal to Capture Network	
Packets	279
Ethereal Options	281
Ethereal Filters	283
Configuring Ethereal and Capturing Packets	283
Viewing Network Traffic between Hosts Using	
EtherApe	288
Configuring EtherApe and Viewing Network	
Traffic	289
Summary	293
Solutions Fast Track	294
Frequently Asked Questions	296

Learn the Flags Used in TCP Connections



Flag	Description
SYN	Synchronize sequence numbers. Used for connection establishment.
FIN	The sender is finished with the connection. Used for connection termination.
RST	Reset the connection.
PSH	Push the data.
ACK	Acknowledgment
URG	Urgent

Chapter 6 Network Authentication and Encryption **299**

Introduction	300
Understanding Network Authentication	300

Answer Your Questions about Kerberos

Q: I wish to remove a principal from the keytab of one of my Kerberos clients. How do I do this?

A: Enter `kadmin` as an administrative user on the Kerberos client (not the KDC) and use the `ktremove` option. For example, if you wanted to remove the principal for the user named *james*, you would do the following:

```
terminal$ /usr/
kerberos/sbin/kadmin
kadmin: ktremove
-p james
kadmin: quit
terminal$
```

Attacking Encrypted Protocols	301
Creating Authentication and Encryption Solutions	303
Implementing One-Time Passwords (OTP and OPIE)	305
What Files Does OPIE Replace?	305
How Does OPIE Work?	305
OPIE Files and Applications	306
<code>opiepasswd</code>	307
Password Format	308
Using <code>opiekey</code>	309
Using <code>opieinfo</code> and <code>opiekey</code> to Generate a List	310
Installing OPIE	310
Configuration Options	310
Installation Options	311
Uninstalling OPIE	312
Exercise: Installing OPIE	312
Exercise: Installing the OPIE Client on a Remote Server	315
Exercise: Using <code>opie-tk</code> and Allowing Windows Users to Deploy OPIE.	316
Exercise: Installing <code>opieftpd</code>	318
Implementing Kerberos Version 5	319
Why Is Kerberos Such a Big Deal?	320
Kerberos Terms	321
Kerberos Principals	322
The Kerberos Authentication Process	323
How Information Traverses the Network	324
Creating the Kerberos Database	325
Using <code>kadmin.local</code>	325
Using <code>kadmin</code>	326
Using <code>kadmin</code> on the Client	328
Using <code>kadmin</code> and Creating Kerberos Client Passwords	329
Setting Policies	330
Using <code>Kinit</code>	330

The kinit Command and Time Limits	332
Managing Kerberos Client Credentials	333
The kdestroy Command	333
Exercise: Configuring a KDC	334
Establishing Kerberos Client Trust Relationships	
with kadmin	337
Additional Daemon Principal Names	339
Logging On to a Kerberos Host Daemon	340
Common Kerberos Client Troubleshooting	
Issues and Solutions	340
Kerberos Client Applications	341
Kerberos Authentication and klogin	342
Exercise: Configuring a Kerberos Client	342
Summary	345
Solutions Fast Track	345
Frequently Asked Questions	348

Chapter 7 Avoiding Sniffing Attacks through Encryption 353

Secure E-Commerce Transactions

If hackers were alerted to an unsecure server, they could capture packets going in and out of the server to gain the data they sought. For example, if an e-commerce server does not use any type of network encryption for transactions, there is a great deal of data to be gained by a hacker. Unfortunately, many small companies or entrepreneurs set up their own Web servers, unaware of potential security problems, and set up simple scripts to process payment forms.

Introduction	354
Understanding Network Encryption	354
Capturing and Analyzing Unencrypted	
Network Traffic	355
Using OpenSSH to Encrypt Network Traffic	
between Two Hosts	361
The OpenSSH Suite	362
Installing OpenSSH	364
Configuring SSH	367
How SSH Works	368
Insecure r-command Authentication	368
Secure SSH Authentication	371
Implementing SSH to Secure Data Transmissions	
over an Insecure Network	373
Distributing the Public Key	376
Capturing and Analyzing Encrypted Network	
Traffic	381
Summary	385

Solutions Fast Track	386
Frequently Asked Questions	388

Chapter 8 Creating Virtual Private Networks 391

Introduction	392
Secure Tunneling with VPNs	392
Telecommuter VPN Solution	392
Router-to-Router VPN Solution	394
Host-to-Host VPN Solution	395
Tunneling Protocols	395
Explaining the IP Security Architecture	396
Using IPSec with a VPN Tunneling Protocol	400
Internet Key Exchange Protocol	401
Creating a VPN by Using FreeS/WAN	402
Downloading and Unpacking FreeS/WAN	404
Compiling the Kernel to Run FreeS/WAN	407
Recompiling FreeS/WAN into the New Kernel	417
Configuring FreeS/WAN	420
Testing IP Networking	420
Configuring Public Key Encryption for Secure Authentication of VPN Endpoints	424
Starting the Tunnel	434
Capturing VPN Tunnel Traffic	436
Closing the VPN Tunnel	438
Summary	439
Solutions Fast Track	440
Frequently Asked Questions	441

Secure Tunneling with Virtual Private Networks (VPNs)



VPNs provide a private data network over public telecommunication infrastructures, such as the Internet, by providing authentication and encryption through a data “tunnel” between devices. All data transmitted between the devices through the tunnel is secure, regardless of what programs the devices are running.

Chapter 9 Implementing a Firewall with Ipchains and Iptables 445

Introduction	446
Understanding the Need for a Firewall	447
Building a Personal Firewall	449
Understanding Packet Filtering Terminology	450

Understand Essential Linux Firewall Functions

- IP address conservation and traffic forwarding
- Network differentiation
- Protection against denial-of-service, scanning, and sniffing attacks
- IP and port
- Content filtering
- Packet redirection
- Enhanced authentication and encryption
- Supplemented logging

Choosing a Linux Firewall Machine	452
Protecting the Firewall	452
Deploying IP Forwarding and Masquerading	453
Masquerading	456
Configuring Your Firewall to Filter Network Packets	458
Configuring the Kernel	460
Packet Accounting	460
Understanding Tables and Chains in a Linux Firewall	461
Built-In Targets and User-Defined Chains	462
Specifying Interfaces	463
Setting Policies	464
Using Ipchains to Masquerade Connections	467
Iptables Masquerading Modules	468
Using Iptables to Masquerade Connections	468
Iptables Modules	470
Exercise: Masquerading Connections	
Using Ipchains or Iptables	471
Logging Packets at the Firewall	471
Setting Log Limits	472
Adding and Removing Packet Filtering Rules	472
ICMP Types	473
Exercise: Creating a Personal Firewall and Creating a User-Defined Chain	475
Redirecting Ports in Ipchains and Iptables	477
Configuring a Firewall	478
Setting a Proper Foundation	478
Creating Anti-Spoofing Rules	479
Counting Bandwidth Usage	483
Listing and Resetting Counters	484
Setting Type of Service (ToS) in a Linux Router	484
Setting ToS Values in Ipchains and Iptables	486
Using and Obtaining Automated Firewall Scripts and Graphical Firewall Utilities	488

Firewall Works in Progress	490
Exercise: Using Firestarter to Create a Personal Firewall	490
Exercise: Using Advanced Firestarter Features	498
Summary	500
Solutions Fast Track	500
Frequently Asked Questions	505

Chapter 10 Deploying the Squid Web Proxy Cache Server 507

Introduction	508
Benefits of Proxy Server Implementation	508
Proxy Caching	508
Network Address Translation	510
Differentiating between a Packet Filter and a Proxy Server	512
Implementing the Squid Web Proxy Cache Server	513
System Requirements Specific to Proxy Caching	516
Installing Squid	517
Configuring Squid	520
The http_port Tag	522
The Cache_dir Tag	523
The acl Tag	525
The http_access Tag	526
Starting and Testing Squid	528
Configuring Proxy Clients	529
Configuring Netscape Navigator and Lynx	530
Configuring Netscape Navigator	530
Configuring Lynx	532
Configuring Internet Explorer (Optional)	533
Summary	535
Solutions Fast Track	536
Frequently Asked Questions	538

**Configure Squid
with the /etc/squid/
squid.conf file**



Chapter 11 Maintaining Firewalls 543

Introduction	544
Testing Firewalls	544
IP Spoofing	546
Open Ports/Daemons	546
Monitoring System Hard Drives, RAM, and Processors	547
Suspicious Users, Logins, and Login Times	547
Check the Rules Database	548
Verify Connectivity with Company Management and End Users	548
Remain Informed Concerning the Operating System	549
Port Scans	549
Using Telnet, Ipchains, Netcat, and SendIP to Probe Your Firewall	550
Ipchains	551
Telnet	551
Using Multiple Terminals	552
Netcat	552
Sample Netcat Commands	554
Additional Netcat Commands	555
Exercise: Using Netcat	557
SendIP: The Packet Forger	558
SendIP Syntax	558
Exercise: Using SendIP to Probe a Firewall	560
Understanding Firewall Logging, Blocking, and Alert Options	563
Firewall Log Daemon	563
Obtaining Firelogd	563
Syntax and Configuration Options	563
Message Format	564
Customizing Messages	566
Reading Log Files Generated by Other Firewalls	568

See How to Use the Firelogd Program



Firelogd (Firewall Log Daemon) is a relatively simple program that can either be run as an application or (you might have guessed) as a daemon. It does two things:

- It reads the kernel log entries and passes them into a "first in, first out" (FIFO) pipe, which *Firelogd* can then process.
- Once its buffer is full, it e-mails a report of suspicious traffic to an account of your choosing. You can have it mailed to a local account, or to a remote system of your choice.

Exercise: Configuring and Compiling Firelogd	568
Fwlogwatch	569
Fwlogwatch Modes	570
Fwlogwatch Options and Generating Reports	572
Exercise: Generating an HTML-Based Firewall Log with Fwlogwatch	575
Automating Fwlogwatch	575
The Fwlogwatch Configuration File	576
Notification Options	579
Response Options	581
Exercise: Configuring Fwlogwatch to Send Automatic Alerts and Block Users	583
Using Fwlogwatch with CGI Scripts	584
Obtaining More Information	586
Viewing the Results	587
Exercise: Using Cron and Fwlogwatch CGI Scripts to Generate an Automatic HTML Report	588
Additional Fwlog Features	590
Obtaining Additional Firewall Logging Tools	590
Summary	593
Solutions Fast Track	593
Frequently Asked Questions	597
Appendix A Bastille Log	599
Appendix B Hack Proofing Linux Fast Track	605
Index	637