



Fundamentals and Implementation

StoneGate Course Handbook

StoneGate Version 2.0

Copyright © 2001–2002 Stonesoft Corp. All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without permission in writing from the publisher.

Stonesoft Corporation	Stonesoft Inc.	Stonesoft Corp.
Itälahdenkatu 22 A	South Terraces, Suite 1000	90 Cecil Street, #11-01
FIN-00210 Helsinki	115 Perimeter Center Place	069531 Singapore
Finland	Atlanta, GA 30346	USA

Trademarks

Stonesoft, the Stonesoft logo, StoneBeat, FullCluster, SecurityCluster, ServerCluster, StoneGate, and WebCluster are trademarks or registered trademarks of Stonesoft Corporation in the United States and/or other countries.

Sun™, Sun™ Microsystems, the Sun™ Logo, Solaris™, and Java™ are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries. All SPARC™ trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the United States and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

Windows®, Windows NT®, and Microsoft® are trademarks or registered trademarks of Microsoft Corporation in the United States and/or other countries.

Linux™ is a registered trademark of Linus Torvalds.

Syntax™ is a registered trademark of Linotype-Hell AG and/or its subsidiaries.

Any other trademarks referenced in this document are property of their respective owners.

Disclaimer

Although every precaution has been taken to prepare this document, Stonesoft assumes no responsibility for errors, omissions, or resulting damages from the use of the information contained in this document. All IP addresses in this document were chosen at random and are used for illustrative purposes only. They are not intended to represent the IP addresses of any specific individual or organization.

Table of Contents

	Preface	xvii
	Comments and Questions	xviii
	Product Export Restrictions	xviii
	Patent Notice	xviii
	Additional Software Licensing Information	xix
	Typographical Conventions	xxix
	Note	xxix
	Caution	xxix
	Tip	xxx
	Typesetting	xxx
	About This Document	xxx
	Course Goals	xxx
	Course Objectives	xxx
CHAPTER 1	General Firewall Principles	1
	Objectives	1
	The Role of the Firewall	1
	Hazards of Networking	2
	The Firewall as Protection	3
	Example Solution	5
	Firewall Technologies	6
	Packet Filters	6
	Proxy Firewalls	8
	Stateful Inspection	9
	StoneGate and Multi-Layer Inspection	11

Firewall Functions	13
Access Control	13
Monitoring and Logging	13
Network Address Translation (NAT)	13
Authentication	14
Virtual Private Networks (VPN)	14
Content Screening	15
Requirements for Modern Firewalls	16
High Availability	16
Scalability	17
High Throughput	17
Centralized Management	18
Firewall Weaknesses	18
Lack of Administration	18
Internal Attacks	19
Summary	19
Review Questions	21
CHAPTER 2	
StoneGate Architecture	23
Objectives	23
General Architecture	24
GUI	25
Management System	27
Firewall Engines	30
Architecture Benefits	32
Distributed Management	32
Virtual LANs	33
High Availability Technologies	35
Built-In High Availability	36
High Availability Connections	38

	Multi-Link Technology Applied	41
	Multi-Layer Inspection	45
	How StoneGate Examines Packets	47
	StoneGate Administration	48
	Administrator Levels	49
	Summary	49
	Review Questions	51
CHAPTER 3	GUI Overview and Implementation Design	53
	Objectives	53
	StoneGate Graphical User Interface (GUI)	54
	Tree View	54
	Status Display	55
	Launchpad	57
	Customizing the Display	61
	Installation Overview	61
	Implementation Strategies	62
	Summary	64
	Review Questions	65
CHAPTER 4	Routing and Anti-Spoofing	67
	Objectives	67
	Firewall Routing	68
	Routing Protocols	69
	StoneGate Routing	70
	IP Spoofing	72
	Advanced StoneGate Routing	75

	Policy Routing Entries.....	75
	Static IP Multicast Routing	77
	Summary.....	79
	Review Questions.....	80
CHAPTER 5	Creating Basic Policies.....	81
	Objectives	81
	Network Elements.....	82
	Host.....	82
	Router.....	82
	Servers.....	83
	Traffic Handlers	84
	Network.....	85
	Firewalls	85
	Group.....	85
	Expression	85
	Address Range.....	86
	Alias	86
	Rules	86
	Access Rules	87
	NAT Rules.....	91
	Summary.....	92
	Review Questions.....	93
CHAPTER 6	Basic Log Management	95
	Objectives	95
	Basic Log Management Theory.....	95
	Fundamental Concepts.....	96
	Logging Options	97

	Alert	98
	Stored	98
	Essential.....	98
	Transient	99
	None	99
	Log Accounting	99
	Log Management in Practice	99
	Log Browser	99
	Filtering Profile Manager.....	101
	Log Pruning Filter Manager	101
	Log Data Manager.....	102
	Summary	103
	Review Questions	104
CHAPTER 7	Administrator Management	105
	Objectives.....	105
	Administrator Accounts	106
	Issues to Consider	106
	Superuser, Editor and Operator	107
	Elements	109
	Simple Elements.....	109
	Granted Elements	110
	Permission Checking	111
	Summary	116
	Review Questions	117
CHAPTER 8	Network Address Translation (NAT)	119
	Objectives.....	119
	Network Address Translation Overview	120

Static Source Translation	120
Dynamic Source Translation.....	122
Destination Translation.....	123
Network Address Translation Examples.....	124
Static Source Example.....	125
Dynamic Source Example	126
Destination Translation Example	127
Outbound Load Balancing NAT.....	128
Proxy ARP (Address Resolution Protocol) and NAT.....	129
Summary.....	130
Review Questions.....	131

CHAPTER 9

StoneGate User Authentication	133
Objectives	133
Introduction to Authentication.....	133
Something They Are	134
Something They Know	135
Something They Have.....	136
StoneGate User Authentication.....	136
LDAP Directories.....	137
How Authentication Is Performed	140
Authentication Types	140
Firewall-Initiated Authentication	141
Client-Initiated Authentication.....	143
Authentication Methods.....	144
Internal Authentication	145
External Authentication.....	145
Summary.....	146

	Review Questions	148
CHAPTER 10	VPN Fundamentals	149
	Objectives	149
	Introduction to VPN	149
	Overview of Cryptography	152
	Symmetric Encryption	153
	Asymmetric Encryption	155
	Diffie-Hellman Key Agreement	157
	Authentication and Integrity	158
	Digital Signatures	159
	Digital Certificates	161
	IPsec Overview	162
	Authentication Header (AH).....	163
	Encapsulating Security Payload (ESP)	164
	Tunnel Mode and Transport Mode	164
	Combination of AH and ESP	165
	Internet Key Exchange (IKE).....	166
	Manual IPsec Mode	170
	Building VPNs with StoneGate	170
	Security Gateways and Sites	170
	Tunnels and Connections	174
	VPN Policy Parameters	175
	Symmetric Encryption Parameters	176
	Data Integrity and Authentication Parameters	177
	Diffie-Hellman Parameters	178
	Perfect Forward Secrecy.....	178
	Lifetime Parameters	179
	Path MTU Discovery.....	180

Hybrid Authentication	180
StoneGate and Certificate Management	180
Certificate Requests	182
Summary.....	183
Review Questions.....	184

LABS

LAB 1	StoneGate Installation	187
LAB 2	Configuring Routing and Anti-Spoofing	223
LAB 3	Creating Basic Policies	231
LAB 4	Basic Log Management	241
LAB 5	Administrator Management.....	249
LAB 6	Network Address Translation (NAT).....	255
LAB 7	Basic User Authentication	265
LAB 8	VPN Fundamentals	275

APPENDICES

	Glossary	291
APPENDIX A	Guidelines for Building Network Security.....	331

APPENDIX B	Multicasting	343
APPENDIX C	References	355

Preface

Welcome to the *StoneGate Fundamentals and Implementation* course for Stonesoft's StoneGate™ High Availability Firewall and VPN solution! StoneGate provides the first fully scalable, high security and high performance firewall and VPN solution for business critical applications. StoneGate is also the first firewall to provide secure connections and load balancing between multiple ISPs to ensure continuous network connectivity.

In addition to this text, you may find it useful to periodically refer to the *StoneGate Administrator's Guide*, which is available through the on-line help system.

Comments and Questions

Please let us know of any errors you find, as well as suggestions for future editions, comments, etc. by writing, Attention Documentation Services, to:

Stonesoft Corporation
Itälahdenkatu 22 A
FIN-00210 Helsinki
Finland

Stonesoft Inc.
South Terraces, Suite 1000
115 Perimeter Center Place
Atlanta, GA 30346 USA

or e-mailing *training@stonesoft.com*.

For technical support information about the StoneGate product, send e-mail to: *support@stonesoft.com*.

For sales questions, information or comments on the StoneGate product, send e-mail to: *info@stonesoft.com*.

For more information, you can also visit our Web site at:
<http://www.stonesoft.com/>

Product Export Restrictions

The products described in this document are subject to export control under the laws of Finland and the European Council Regulation (EC) N:o 1334/2000 of 22 June 2000 setting up a Community regime for the control of exports of dual-use items and technology (as amended). Thus, the export of this Stonesoft software in any manner is restricted and requires a license by the relevant authorities.

Patent Notice

Multi-Link, Multi-Link VPN, and the StoneGate clustering technology—as well as other technologies included in StoneGate—are protected by pending patent applications in the U.S. and other countries.

Additional Software Licensing Information

The StoneGate software includes several open source or third-party software packages to support certain features. This section provides the appropriate software licensing information for those products.

GNU General Public License

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:
 - a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
 - b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
 - c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

-
3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:
 - a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
 - b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
 - c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.
5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.
6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.
7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the

Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program. If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.
9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.
12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR

INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author

Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.

This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

OpenSSL Toolkit

This software includes the OpenSSL toolkit.

LICENSE ISSUES

=====

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact openssl-core@openssl.org

OpenSSL License

Copyright (c) 1998-2000 The OpenSSL Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

All advertising materials mentioning features or use of this software must display the following acknowledgment:

“This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (<http://www.openssl.org/>)”

The names “OpenSSL Toolkit” and “OpenSSL Project” must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.

Products derived from this software may not be called “OpenSSL” nor may “OpenSSL” appear in their names without prior written permission of the OpenSSL Project.

Redistributions of any form whatsoever must retain the following acknowledgment: “This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (<http://www.openssl.org/>)”

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT “AS IS” AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes cryptographic software written by Eric Young, (ey@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

Original SSLeay License

Copyright (C) 1995-1998 Eric Young (ey@cryptsoft.com). All rights reserved.

This package is an SSL implementation written by Eric Young (ey@cryptsoft.com). The implementation was written so as to conform with Netscape's SSL. This library is free for commercial and non-commercial use as long as the following conditions are adhered to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is 'Tim Hudson' (tjh@cryptsoft.com). Copyright remains Eric Young's, and as such any Copyright notices in the code are not to be removed. If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used. This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

All advertising materials mentioning features or use of this software must display the following acknowledgement: "This product includes cryptographic software written by Eric Young (ey@cryptsoft.com)" The word 'cryptographic' can be left out if the routines from the library being used are not cryptographic related :-).

If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement: "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The licence and distribution terms for any publically available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution licence [including the GNU Public Licence.]

OpenLDAP

This software includes the OpenLDAP client developed by The OpenLDAP Foundation. Original version of the OpenLDAP client can be downloaded from <http://www.openldap.org>

This software includes the OpenLDAP server.

The OpenLDAP Public License
Version 2.7, 7 September 2001

Redistribution and use of this software and associated documentation ("Software"), with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain copyright statements and notices,
2. Redistributions in binary form must reproduce applicable copyright statements and notices, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution, and
3. Redistributions must contain a verbatim copy of this document.

The OpenLDAP Foundation may revise this license from time to time. Each revision is distinguished by a version number. You may use the Software under terms of this license revision or under the terms of any subsequent revision of the license.

THIS SOFTWARE IS PROVIDED BY THE OPENLDAP FOUNDATION AND CONTRIBUTORS "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OPENLDAP FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

OpenLDAP is a trademark of the OpenLDAP Foundation.

Copyright 1999-2001 The OpenLDAP Foundation, Redwood City, California, USA. All Rights Reserved. Permission to copy and distributed verbatim copies of this document is granted.

libradius1

This software includes the `libradius1` package.

Copyright (C) 1995,1996,1997,1998 Lars Fenneberg <lf@elemental.net>

Permission to use, copy, modify, and distribute this software for any purpose and without fee is hereby granted, provided that this copyright and permission notice appear on all copies and supporting documentation, the name of Lars Fenneberg not be used in advertising or publicity pertaining to distribution of the program without specific prior permission, and notice be given in supporting documentation that copying and distribution is by permission of Lars Fenneberg.

Lars Fenneberg makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

Copyright 1992 Livingston Enterprises, Inc.

Livingston Enterprises, Inc. 6920 Koll Center Parkway Pleasanton, CA 94566

Permission to use, copy, modify, and distribute this software for any purpose and without fee is hereby granted, provided that this copyright and permission notice appear on all copies and supporting

documentation, the name of Livingston Enterprises, Inc. not be used in advertising or publicity pertaining to distribution of the program without specific prior permission, and notice be given in supporting documentation that copying and distribution is by permission of Livingston Enterprises, Inc. Livingston Enterprises, Inc. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

[C] The Regents of the University of Michigan and Merit Network, Inc. 1992, 1993, 1994, 1995 All Rights Reserved.

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies of the software and derivative works or modified versions thereof, and that both the copyright notice and this permission and disclaimer notice appear in supporting documentation.

THIS SOFTWARE IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE REGENTS OF THE UNIVERSITY OF MICHIGAN AND MERIT NETWORK, INC. DO NOT WARRANT THAT THE FUNCTIONS CONTAINED IN THE SOFTWARE WILL MEET LICENSEE'S REQUIREMENTS OR THAT OPERATION WILL BE UNINTERRUPTED OR ERROR FREE. The Regents of the University of Michigan and Merit Network, Inc. shall not be liable for any special, indirect, incidental or consequential damages with respect to any claim by Licensee or any third party arising from use of the software.

Copyright (C) 1991-2, RSA Data Security, Inc. Created 1991. All rights reserved.

License to copy and use this software is granted provided that it is identified as the "RSA Data Security, Inc. MD5 Message-Digest Algorithm" in all material mentioning or referencing this software or this function.

License is also granted to make and use derivative works provided that such works are identified as "derived from the RSA Data Security, Inc. MD5 Message-Digest Algorithm" in all material mentioning or referencing the derived work. RSA Data Security, Inc. makes no representations concerning either the merchantability of this software or the suitability of this software for any particular purpose. It is provided "as is" without express or implied warranty of any kind.

These notices must be retained in any copies of any part of this documentation and/or software.

TACACS+ Client

This software contains TACACS+ client.

Copyright (c) 1995-1998 by Cisco systems, Inc.

Permission to use, copy, modify, and distribute this software for any purpose and without fee is hereby granted, provided that this copyright and permission notice appear on all copies of the software and supporting documentation, the name of Cisco Systems, Inc. not be used in advertising or publicity pertaining to distribution of the program without specific prior permission, and notice be given in supporting documentation that modification, copying and distribution is by permission of Cisco Systems, Inc.

Cisco Systems, Inc. makes no representations about the suitability of this software for any purpose. THIS SOFTWARE IS PROVIDED “AS IS” AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

MD5C.C - RSA Data Security, Inc., MD5 message-digest algorithm

Copyright (C) 1991-2, RSA Data Security, Inc. Created 1991. All rights reserved.

License to copy and use this software is granted provided that it is identified as the “RSA Data Security, Inc. MD5 Message-Digest Algorithm” in all material mentioning or referencing this software or this function.

License is also granted to make and use derivative works provided that such works are identified as “derived from the RSA Data Security, Inc. MD5 Message-Digest Algorithm” in all material mentioning or referencing the derived work.

RSA Data Security, Inc. makes no representations concerning either the merchantability of this software or the suitability of this software for any particular purpose. It is provided “as is” without express or implied warranty of any kind.

These notices must be retained in any copies of any part of this documentation and/or software.

Typographical Conventions

The following typographical conventions are used throughout this material to aid the reader’s comprehension:

TABLE 0.1 *Typographical Conventions*

Formatting	Informative Uses
Garamond	Text body.
Garamond Bold	Interface elements (buttons, menus, icons) and any other interaction with the user interface.
<i>Garamond Italics</i>	Commands, programs, etc. Also for book titles, cross-references and first use of acronyms or terms.
Courier	File names, directories, code or scripts displayed on screen.
Courier Bold	User input of commands, code, or scripts.
<i>Courier Italics</i>	Variables that should be substituted for other values.

In addition to those typographical conventions, the following icons are used to indicate additional information of value to the reader:



Note

The note icon represents additional information that the user should read.



Caution

The caution icon represents cautionary, or critical information that the user should take into advisement before performing an action or implementing a feature.



Tip

The tip icon is for ideas on ways to perform actions, suggestions on implementation or configuration, and other ideas that may be considered, but are not necessary, to use StoneGate.

Typesetting

This document was produced with Adobe[®] FrameMaker[®], and is set in Adobe[®] Garamond[®], FF City Streets, FF Trademark, Syntax[®], and Adobe[®] Berthold Akzidenz Grotesk.

About This Document

This course handbook is divided into three main parts. The first section, immediately following this preface, contains the detailed chapters, where the theory and examples are presented. The second section consists of the lab exercises that can be performed in a StoneGate classroom, and the third section contains reference appendices with supplemental information.

Course Goals

In this course, *StoneGate Fundamentals and Implementation*, the student will learn the basics of firewalls, including where and why firewalls are used. StoneGate as a firewall technology will then be discussed, with an overview of the architecture of StoneGate being presented and contrasted with alternative firewall technologies. Installation of StoneGate, including considerations for the management system, hardware requirements, and network components will then be covered. Students will create network elements (define a network environment), define services (protocols and ports), create basic rules, and build a basic VPN between firewalls. At the end of the course the student will have a familiarity with clustering the firewall nodes, and an overview of the advanced features of StoneGate that will be covered, in detail, in the *StoneGate Advanced Implementation and Beyond* course.

Course Objectives

Upon completion of the *StoneGate Fundamentals and Implementation* course, students should be able to:

- explain the basics of firewall technologies
- explain, in detail, the architecture of StoneGate
- install and configure a basic StoneGate firewall
- describe the purpose of each manager in the user interface
- create basic rules and network elements in the interface
- add services, address translation, alerts, and basic rule options
- add users in the internal user directory for user authentication
- establish a basic VPN between firewalls (using IPsec)
- pass the *Certified StoneGate Engineer (CSGE)* examination.

General Firewall Principles

Network security breaches can be very embarrassing, especially when they happen to organizations that should have known better.

– William Cheswick, Steven Bellovin

This unit introduces and discusses the underlying security principles of firewalls. In this chapter you will learn what firewalls are, which different types of firewalls there are, how they are used, what they are capable of, as well as what their possible weaknesses are.

Objectives

Upon completing this unit, you should be able to:

- explain the purpose of a firewall
- list three main functions of a firewall
- describe four different types of firewall technologies
- explain at least two capabilities of firewalls
- summarize the weaknesses of firewalls.

The Role of the Firewall

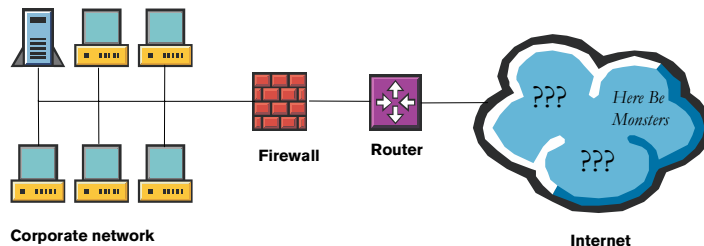
It is difficult to give an all-inclusive, yet simple definition of a firewall, but here we shall present some outlines that should help form a picture of features and functions they have. First, we'll take a look at which types of threats—in addition to the opportunities—the modern network environment contains, and how firewalls can respond to these concerns.

Hazards of Networking

The way the Internet can be seen today, both as an opportunity and a threat, is in some ways comparable to how the explorers and sailors of the 16th century viewed the still uncharted oceans: they implied both potential routes to unforeseen fortunes and, at the same time, looming perils in the form of pirates, competing powers, or—who knows—some unknown sea monsters.

Today, as corporations are becoming more and more dependent on their Internet connections, they are also becoming more and more frequently attacked from the outside by way of exploiting those very connections; be it by adventurous hackers or by professional information thieves. In general, it could be said that there is no rhyme or reason to network attacks, and no network is small enough to be ignored and protected by a mere *security through obscurity* strategy. Any system can become the target of an attacker—for example, Microsoft, eBay, and the New York Yankees were victims of a widely publicized attack in October 2000. Smaller organizations are raided frequently with less public commotion. Attackers may grab data to sell as mailing lists, to use for credit or financial access, to add to corporate profiles, or to cripple competitors. They may also destroy data for many reasons including vindictiveness, an urge to show off, or boredom.

FIGURE 1.1 *Uncharted threats of the Internet*



In addition to such external attacks, *internal* network attacks are also an issue. But one must bear in mind that it's beyond the scope of firewalls to give protection against attacks by authorized persons located within the protected network. Other aspects of the corporate security policy should cover that type of risk.

You don't want unauthorized parties to get their hands on confidential data. You also want to ensure that information remains intact, so that no one can alter crucial data unnoticed. And of course, you want your data and resources to be available whenever needed. These three parameters—*confidentiality*, *integrity*, and *availability*—are the main goals of any credible security policy. Firewalls can help to achieve these objectives in the network environment by blocking unauthorized persons from having access to sensitive data, computer memory and processing power.

The Firewall as Protection

Because of possible intruders and information spies, an open corporate network implies intolerable risks. The advantages of the Internet in communication and business cannot be attained without proper protection from hostile attackers and without confidence that private information remains private. Firewalls are a cornerstone in the defense strategy that aims at eliminating the misuse of confidential data and resources by any unauthorized parties. A firewall is just one, albeit important, element in the overall corporate security policy. Its role is to implement that policy in the network environment.

Firewalls regulate communication on data networks; or more precisely, between networks with different security levels. Their main purpose is to control the traffic that passes through from one network to another, and deny access to network resources from undesired or potentially harmful packets and connections.

The principle of access control is ideally expressed as “*whatever is not expressly permitted is denied*.” By default, nobody and nothing should be

permitted entry to the internal network. That means that in order for any traffic to be allowed into the network, it must first satisfy a specifically designed rule that permits limited access. Typically, internal network clients are granted a more unrestricted access to external networks, but also outbound connections need to be controlled.

Some advantages that firewalls can provide include:

- Firewalls can protect the corporate intranet from undesired traffic, including everything from malicious attackers to unsolicited e-mail (spam), on the basis of the corporate security policy implemented by the network administrator.
- Firewalls can secure sensitive corporate information and resources within a LAN. This insulates departments such as Research & Development, or Human Resources from other company departments, limiting the access within any one area to authorized users only.
- Firewalls can concentrate network security policies at a single point. When policies or administrators change, the instructions, configurations, and passwords need only be changed in one place. One must, however, ensure that the firewall doesn't turn out to be a *single point of failure*.

Firewalls can also be used for other purposes, such as monitoring network traffic and the use of network resources, authenticating users, and implementing virtual private networks (VPN). These functions will be covered in more detail in section “*Firewall Functions*” on page 13 and in subsequent chapters.

The firewall offers a reasonable amount of protection against Internet intruders, on the condition that the security policies enforced by the firewalls are carefully designed, and there are no loopholes or back doors. Firewalls can only control traffic that actually passes through them; even the most carefully planned firewall system is undermined

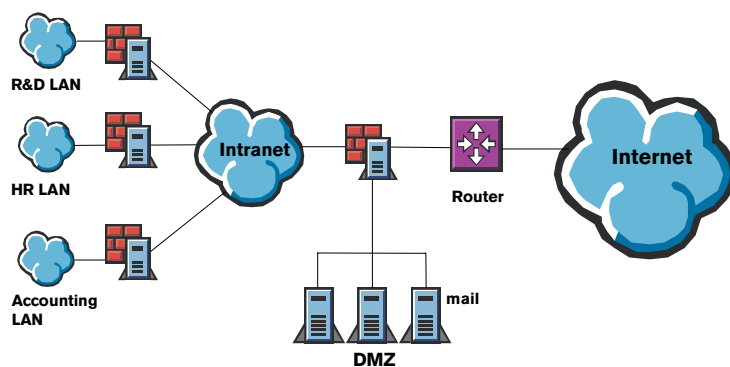
by a single back door—say, a modem connection from the intranet to the Internet—that allows traffic to circumvent the firewall. Please see *Appendix A*, “Guidelines for Building Network Security” for background information on designing network security.

Example Solution

For example, in Figure 1.2, the *demilitarized zone* (DMZ), containing a publicly available pool of servers from which clients can reach resources from the Internet, is protected by a firewall. Other firewalls guard particularly sensitive networks within the company intranet, such as those of the Research and Development, Human Resources, and Accounting departments.

This type of solution means there is no direct access from the Internet into the internal network. Anyone trying to access internal resources from the Internet would have to pass through at least one firewall. For example, a company mail server can be located on the firewall-protected DMZ, allowing SMTP-based traffic to that machine. However, stricter access to the HR LAN can be imposed, denying traffic to it from both the DMZ and the Internet by the HR firewall.

FIGURE 1.2 *An example of deployment of firewalls in a corporate network*



Firewall Technologies

On the firewall market there is a wide range of both software-based and hardware-based firewall solutions from lightweight, personal firewalls to scalable enterprise-wide systems. Software-based solutions are usually installed on standard hardware (such as Intel[®], Sun Microsystems SPARC[®], Hewlett-Packard, I.B.M., or equivalent server hardware), whereas hardware-based solutions are typically proprietary in nature. Either way, the firewalls can be categorized by the way they handle network traffic. In this section, we shall give an overview of the existing firewall technologies, and we'll also put on that map a new type of firewall: *StoneGate*.

Traditionally, firewalls can be divided into three main groups:

- packet filtering firewalls
- application-level proxy firewalls
- stateful inspection firewalls.

Next, we will briefly compare each technology, and bring out their fortes and drawbacks. We will also see how StoneGate fits into this picture.

Packet Filters

The packet filtering firewall, based on the header information of the packets it receives, allows or denies access to or from the network that it is guarding. Packet filtering firewalls check each packet and can be implemented by most common routing devices.

Packet filters typically contain *access control lists* (ACLs) to monitor the following header data:

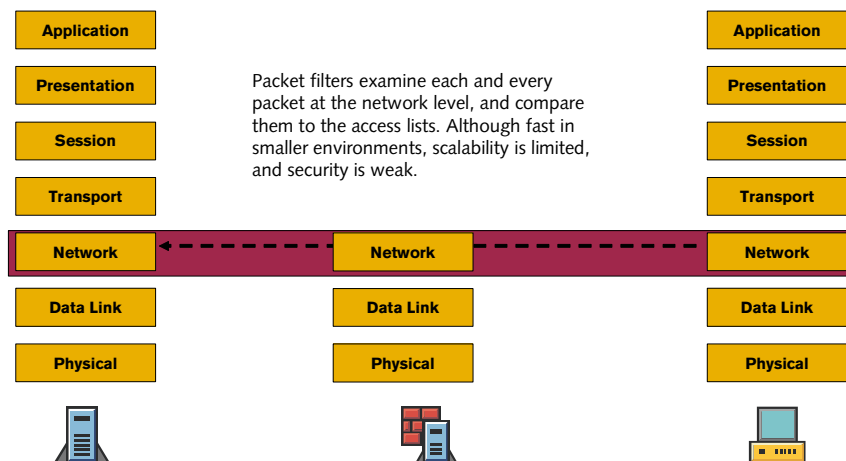
- source IP address
- destination IP address
- source port

- destination port
- ICMP message type
- protocol
- packet size
- various header flags.

By combining these parameters, complex policies can be enforced by the packet filter.

Packet filters tend to have high performance, because their inspection involves very simple parameters at the network layer of the TCP/IP stack. They typically only inspect up to the network layer, as show in Figure 1.3. With more complex environments, however, as every packet of every connection is checked against the access control rules, the performance of packets filters drops significantly.

FIGURE 1.3 *Packet filtering model*

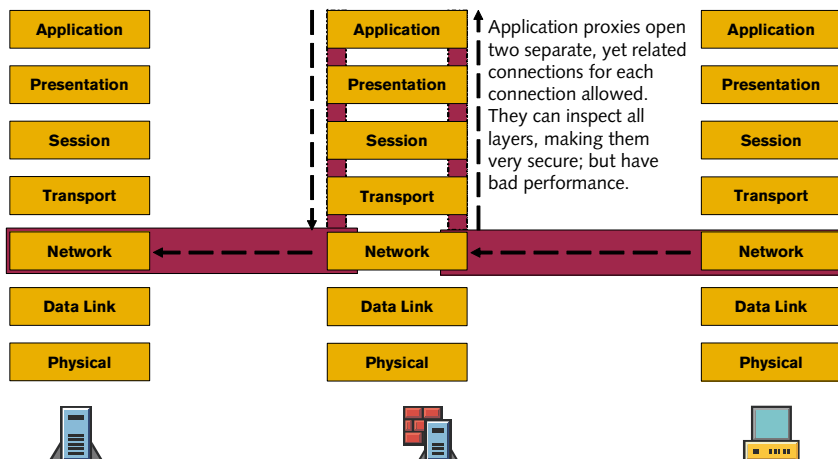


In conclusion, packet filtering is neither very flexible nor generally very secure because the network layer lacks the context of each packet. This type of filtering can be easily fooled with techniques such as fragmented packets, or bogus or invalid IP address information. Packet filters cannot protect against malicious contents in higher levels of the protocol stack, or examine the actual data portion. Thus, they are often used in combination with application-level proxies. Packet filtering is commonly used also in routers at the network perimeters.

Proxy Firewalls

Proxy firewalls are firewalls running application proxy services. This means that the server establishes a second, different connection to the destination network on behalf of the host from the source network—if the packet meets the security policy criteria for it to pass through. In other words, proxy firewalls mediate communications between two different devices located on different networks.

This type of firewall is fully application-aware, and therefore very secure, but at the same time there's a trade-off concerning performance, due to the additional overhead required to maintain separate connections, and to inspect packets up to the application layer.

FIGURE 1.4 *Proxy firewall model*

Proxy firewalls can cache information such as HTML pages in order to gain some extra performance, but the application layer awareness continues to affect performance.

First, application level checking and duplicate connections can drain system resources, affecting firewall performance. Second, the number of services used by a proxy firewall is an issue. Since every service needs its own proxy, the proxy list is always in some sense incomplete, and new applications and services are hard to keep up with.

Traffic inspection takes place at the highest layer of the TCP/IP stack. The necessity of inspecting every layer before returning back down to the physical layer can cause bottlenecks in busy networks.

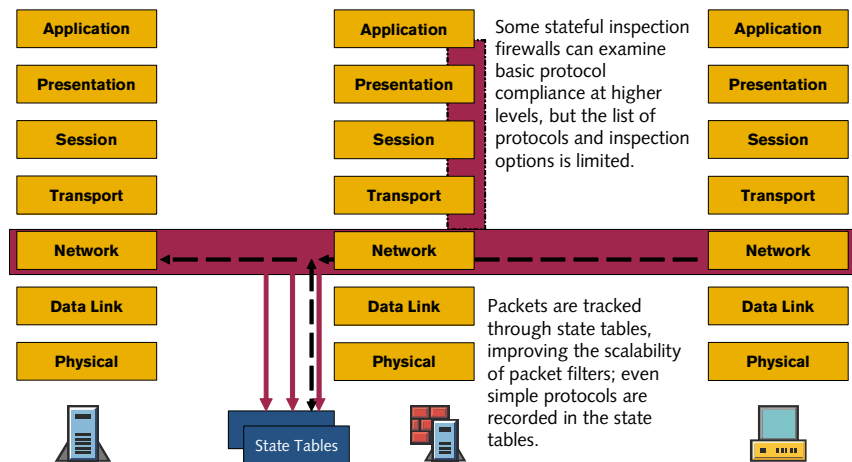
Stateful Inspection

Stateful inspection technology was developed to overcome the limitations of packet filtering firewalls. Stateful inspection firewalls use additional criteria, such as historical data about the connection, in determining whether to allow or deny access. They track the

established connections and their states in *dynamic state tables* and ensure that the connections comply with the security policies.

By applying connection status and context information to current connections and packets, some packets are denied access before being further inspected at a higher level, which increases performance. Furthermore, since stateful inspection understands the context of connections (and therefore can relate the returning packets to appropriate connections), connections already determined to be “secure” can be allowed without further examination. This is especially important with services such as telnet and FTP. Stateful inspection operates just beneath the network layer (in practice, the inspection takes place between the data link layer and network layer). Stateful inspection firewalls have a rather limited capability to inspect data at the application layer. Stateful inspection systems also keep state tables for every connection protocol, whether it is conceivable or not.

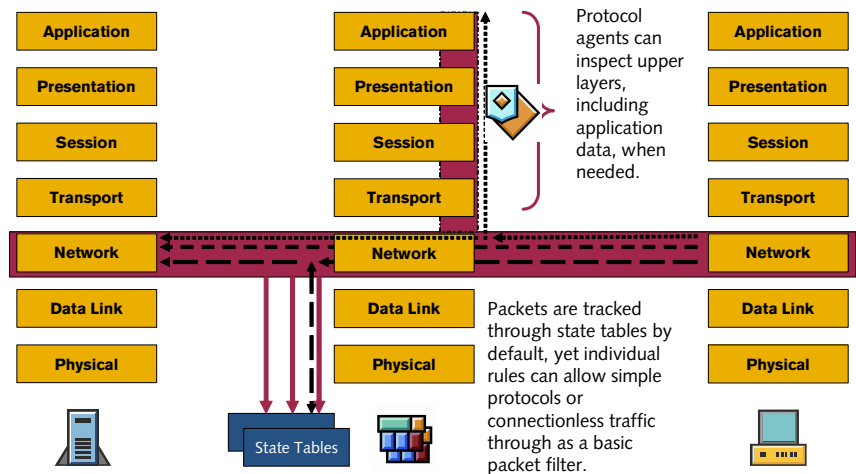
FIGURE 1.5 *Stateful inspection model*



StoneGate and Multi-Layer Inspection

With StoneGate, Stonesoft introduces a new firewall technology called *Multi-Layer Inspection*SM. Like stateful inspection, StoneGate uses state tables to track connections and judge whether a packet is a part of an established connection. However, it also features application-layer inspection by implementing specific *Protocol Agents*, when necessary, for enhanced security. Thus, StoneGate can inspect data all the way up to the application layer to decide whether a packet should be granted access or not. Moreover, StoneGate can also act as a packet filter for types of connections that do not require the security considerations of stateful inspection.

FIGURE 1.6 *Multi-Layer Inspection model*



The state of connections provides valuable information for assessing incoming packets. Any packet must be either accepted directly by the rule base, be a part of a previously accepted connection, or of a related connection. Whenever packets arrive, the firewall checks them against active connections before proceeding through the rules of the security policy. If a connection has a registered state, all the packets following

the opening packet can pass the firewall securely without having to traverse the rule base.

By default, most rules in StoneGate security policies implement stateful inspection methods, but the administrator can flexibly configure rules with simple packet filtering for certain types of traffic. For example, SNMP traps from server systems and network devices can pass through the firewall to a management network on the basis of simple packet filtering in case there's no need to enforce a more strict inspection. This kind of flexibility enhances the firewall performance.

In addition, application level security can be applied to specific rules in the security policy when needed. Multi-Layer Inspection is capable of providing this type of security without the performance degradation of conventional proxy firewalls. StoneGate can implement application level inspection without the need to handle two separate connections. This is achieved with the components called Protocol Agents that can be assigned to certain types of traffic.

Protocol Agents are also used to handle complex connections (e.g., Oracle[®] or FTP), to redirect traffic to content inspection servers, to enforce protocol standards and to modify data payload if necessary. The FTP Protocol Agent, for example, can inspect the control connection and only allow packets containing valid FTP commands. It can also be configured to redirect traffic to a CIS for content screening and to modify IP addresses in the payload in case of network address translation is required. The Protocol Agents are covered in more detail in the *StoneGate Advanced Implementation and Beyond* course.

In brief, Multi-Layer Inspection combines application layer inspection, stateful inspection, and packet filtering technologies flexibly for added security without affecting system performance. In

the next section, the different functions these firewall technologies can perform will be discussed.

Firewall Functions

A firewall can have several different functions on a network. Although their main function is to control network access, firewalls can typically be configured for other basic network security tasks and, additionally, for more complex monitoring and filtering functions.

Access Control

The primary task of any firewall is to control access to data resources, so that only authorized connections are allowed. Access control is enforced in access rules, which are combined into rule bases. The rules collected into rule bases reflect the corporate network security policy.

Monitoring and Logging

Firewalls can be used to measure and monitor traffic load and attributes. A very important firewall feature is the ability to log monitored traffic. Properly recorded log data can be used to detect intruders, and establish evidence to use against attackers. That kind of forensic evidence may prove to be invaluable in case hacking leads to lawsuits. More commonly, logging is used to track the use of network resources, building a case for required services or hardware. Logging also helps administrators detect and troubleshoot network misconfigurations or failures. For more information on logging, please see Chapter 6, *Basic Log Management*, on page 95.

Network Address Translation (NAT)

Network address translation (NAT) is a feature that enables the firewall to modify the IP headers of packets it forwards. It was originally created to alleviate the problem of the rapidly diminishing

IP address space. By changing the network address of the originating (internal) network, your network gains an added side-benefit; the private IP addresses of hosts and the structure of an internal network can be concealed by a firewall. In fact, NAT enables you to hide your entire network behind even a single public IP address.

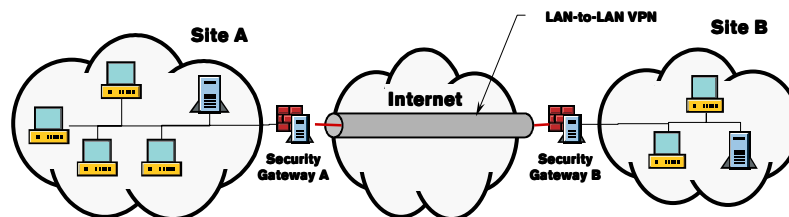
As handy as NAT can be in terms of scarce public IP addresses and security, it is important to understand that NAT is *not* primarily a security feature. It is simply a method of modifying packets that lends itself to security applications. For more information on network address translation, please see Chapter 8, *Network Address Translation (NAT)*, on page 119.

Authentication

Firewalls are often used to authenticate users accessing network resources from other locations. The various authentication methods ensure that the users trying to connect really are who they claim to be. These methods may involve a third-party authentication service based on standard protocols like RADIUS or TACACS+, but it can also be based on the originating IP address or other parameters, such as usernames and passwords. For more information on authentication, please see Chapter 9, *StoneGate User Authentication*, on page 133.

Virtual Private Networks (VPN)

Virtual private networks (VPN) conceal and encrypt traffic between end-points to establish a virtual, secure tunnel through an insecure, typically public, network. Firewalls are used at the tunnel end-points as *security gateways* to encrypt and decrypt data passing between them, creating a *site-to-site VPN*. VPNs can also be established between a client machine, such as a remote laptop and the firewall. Figure 1.7 illustrates a simple, site-to-site VPN.

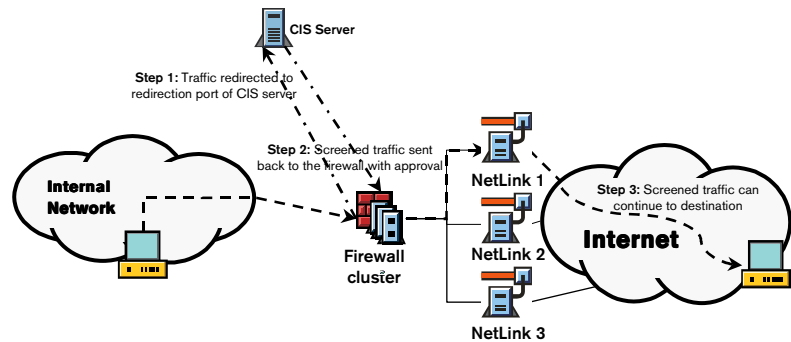
FIGURE 1.7 *Simple VPN*

When a packet leaves Site A, destined for Site B, and reaches gateway A, it is encrypted by the firewall. Once it reaches the Site B gateway, the packet is decrypted and re-shaped into cleartext form. When concealing the traffic this way, it is possible for two different sites that might be on other sides of the world from each other to share resources and communicate with each other safely over the Internet. For more information on virtual private networks, please see Chapter 10, *VPN Fundamentals*, on page 149.

Content Screening

Firewalls are usually not used as the primary tool for virus detection. However, they can be used in combination with content inspection servers (CIS) which check the data portion of incoming packets, such as HTTP or FTP traffic. Anything deemed malicious is either stripped from the packet or the packet is denied access, according to the administrator's security rules. This way, viruses or hazardous content are discarded before packets enter the internal network.

For instance, incoming SMTP e-mail traffic could be forwarded from the firewall to the CIS for virus and content checking. Once any suspicious content is removed, the “scrubbed” packets are returned from the CIS back to the firewall for routing to their final destination.

FIGURE 1.8 *Content screening with CIS*

Another application of content inspection with firewalls controls traffic flow in the opposite direction. The HTTP based Web traffic can be sent from the firewall to a CIS, which can then examine the destination site's address (URL). If the CIS decides the site is on the list of "inappropriate" sites, the traffic is denied. Approved traffic is forwarded as in Figure 1.8.

Requirements for Modern Firewalls

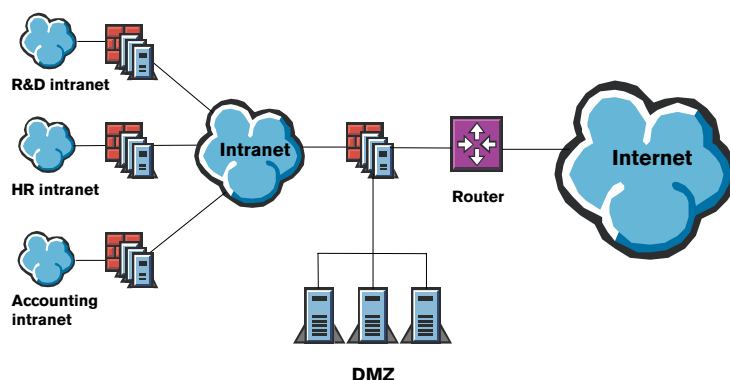
As the volume and the importance of Internet traffic keeps growing, it becomes increasingly essential that the firewalls will continue to meet certain specific requirements. We'll give here an overview of what types of things will be expected from firewalls in the rapidly changing network traffic environment.

High Availability

Advanced clustering technologies prevent firewalls from being either "bottlenecks" or single points of failure. Firewalls are clustered when multiple firewall nodes function as a single, virtual entity and enforce identical security policies. The performance of each node contributes to the total throughput, eliminating bottlenecks, and providing a fault tolerant and reliable firewall. High availability means that traffic from

a failed node can be switched over to other nodes transparently. This also allows for online maintenance of nodes without disturbing the traffic. Securing the availability of your network services is crucial also from the point of view of user satisfaction and, therefore, it very likely will also have economical implications. Clustering can be implemented either with specific additional hardware, with add-on software, or with built-in clustering capability.

FIGURE 1.9 *Eliminating firewalls as single point of failure through clustering*



Scalability

As traffic volumes grow congestion will start to disturb the flow of network traffic sooner or later. The firewall is by definition a “choke point”, through which all traffic should pass. Therefore, it is crucial that the throughput of the firewall will not become the limiting factor for network connections. The possibility to cluster firewalls also means that new firewall nodes can be added flexibly as traffic volumes grow, thereby enhancing the load balancing of traffic.

High Throughput

Gigabit network environments are becoming more and more common, and the firewalls will be required to cope with this

evolution. Clustering of firewall nodes also contributes to better throughput, as does the load balancing of multiple ISP links.

Centralized Management

Geographically widespread, multi-national, and smaller networks all benefit from centrally managed firewall security policies. Corporate security policies often co-exist with site-specific rules to provide the required degree of granularity in the implementation of network security policy. The implementation of corporate-wide, complicated security policies requires a great deal of flexibility from the firewall management system.

Centralized and efficient management of administrator rights can also be seen as a way to minimize the possibility of human error. To avoid unintentional confusion or harm, access to firewall configuration and rule bases should be carefully planned according to the level of authority and expertise of the administrators.

The capability of remote installation and configuration is another important feature that will become more crucial in the evolving complicated and distributed network solutions.

Firewall Weaknesses

Knowing the benefits that firewalls can offer is most useful when you are equally aware of their weaknesses. A balanced approach is essential to form effective corporate security policies. You need to have a good picture of the whole security framework to decide what other measures are needed in addition to firewalls.

Lack of Administration

Complex network environments with multiple Internet interfaces for VPN, remote access, e-business, and cache servers have increased the demand for administrators and administrative skills. This is in part

because firewalls cannot provide effective security without careful attention and maintenance.

A firewall must be thoughtfully installed and configured. Security policies need periodic evaluation and regular updates. Too often, a firewall gathers dust in a corner for years without a professional administrator to look after it. Often, only after an attack and serious or complete data loss occurs, is it recognized that the system needs to be actively administered.

Internal Attacks

Having well-designed and maintained firewalls is definitely a key ingredient to good network security. But firewalls, content inspection servers, and other devices examining packets at the network perimeter are ineffective against *internal* attacks. By some estimates, around 60 percent of all network attacks, including data theft, loss of resources, or destruction of data are launched from within the corporation. These kinds of attacks are much more difficult to defend against, and require different approaches than firewalls or other perimeter defenses. The implementation of the corporate security policy should address these issues through, for example, security training of employees and host-based virus protection.

Summary

A firewall is a gateway between two or more networks. Its main purpose is to control the traffic that passes through it from one network to another and deny the entrance of undesired or potentially harmful packets and connections. Firewalls determine access based on security policies installed by the firewall administrator. The main firewall functions are:

- access control
- authentication

- VPN
- NAT
- monitoring and logging
- content scanning.

Firewall technologies can be categorized by the way they handle traffic:

- packet filtering
- application proxy
- stateful inspection
- Multi-Layer Inspection.

Firewalls do not have to be throughput bottlenecks because multiple firewalls can be configured to act as one, using clustering technologies. This produces high availability, eliminating the firewall as single point of failure.

With all the capabilities, one must bear in mind that firewalls do not protect networks from abuse originating from within the internal network; be it human error, administrator mistakes, or deliberate attacks.

Review Questions

- What are the main functions of a firewall?
- What are the four types of firewalls?
- How is it possible to prevent a firewall from being a single point of failure?
- Which type of security hazards can firewalls not respond to?

StoneGate Architecture

Any sufficiently advanced technology is indistinguishable from magic.

– Arthur C. Clarke

This chapter describes the StoneGate™ architecture. Starting with an overview of the main StoneGate components, the focus shifts to the firewall engine itself, describing how the engine processes packets using administrator-defined security policies and network elements.

In this chapter, you will look at the fundamentals of the clustering technology, load balancing, and the design of the management system, followed by packet handling by the firewall nodes. Finally, an overview of the administrator levels will be presented.

Objectives

Upon completing this unit, you should be able to:

- list the three main components of StoneGate
- explain the difference between “Management Server” and “management system”
- explain the role of each StoneGate component
- describe the clustering/load-balancing features of StoneGate
- list the three administrator levels.

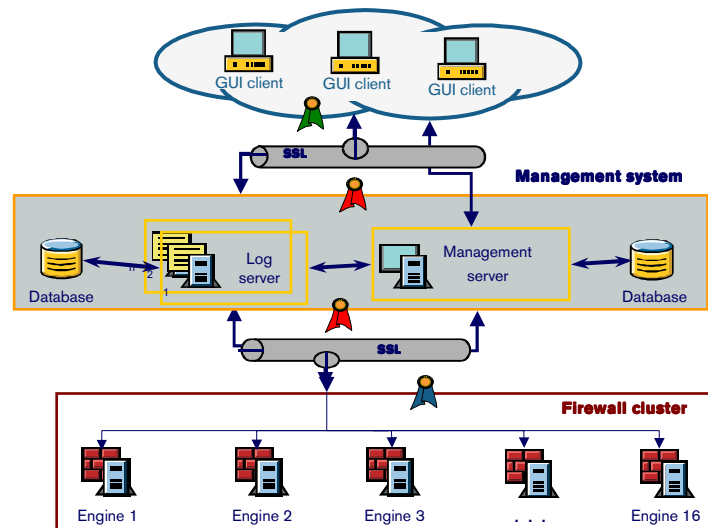
General Architecture

The basic design strategy for StoneGate is a distributed architecture. A StoneGate system consists of three main components:

- the graphical user interface (GUI)
- the management system
- the firewall engines.

As illustrated in Figure 2.1, the management system has additional components, which can also be distributed among several machines. The firewall engines enable scaling from a single machine to a 16-node cluster. Because of this, a StoneGate system enables a secure, high availability enterprise. In this section, we'll examine each of the main components in further detail, discussing the role of each, and how they operate with each other.

FIGURE 2.1 *StoneGate architecture overview*



GUI

The GUI—or the *Administration Client*—is the main interface used to configure and administer all aspects of StoneGate. It is a Java™-based program which allows administrators: to configure the firewall nodes; to describe the networks and systems being protected by creating hosts, servers, services, VPNs, firewalls and firewall clusters, routers and other network devices; and to design the policies to be implemented to protect those systems and networks. In addition, it is used to administer users and authentication services, manage log data, and monitor firewalls and clusters. The GUI communicates with the various components of the management system, representing performance data, log data, configuration information and policies in a user-friendly format.



.....

Note: For security reasons, the GUI does not communicate directly with the firewall engines. All changes to the firewall nodes are handled through the management system components.

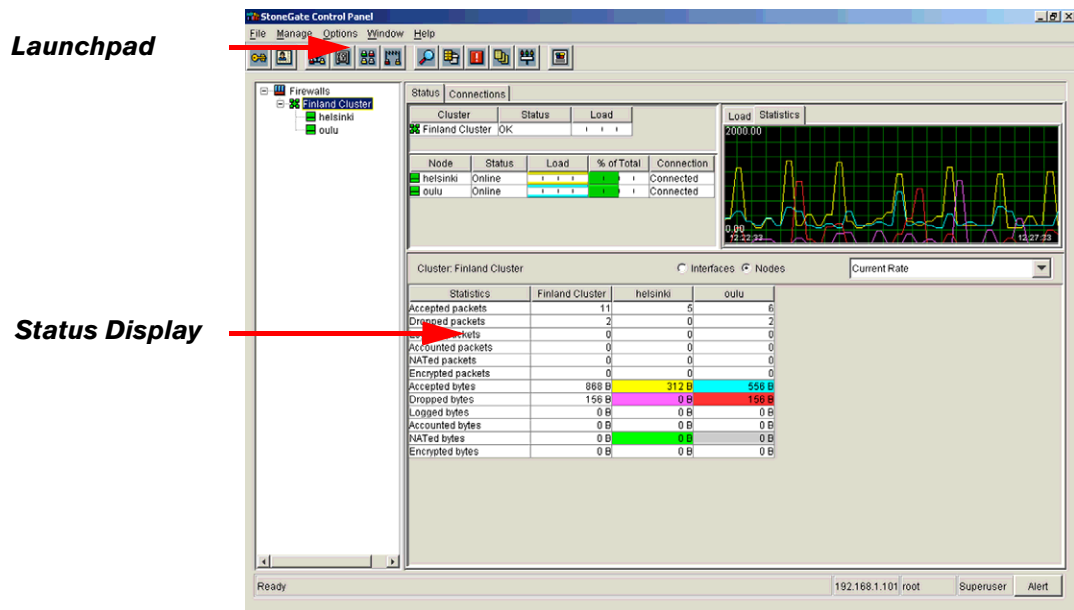
.....

The GUI can be deployed anywhere on the network, and multiple instances of it can be run at once. For example, an administrator can use the GUI to modify the properties of a local network in a branch office, while the senior administrator in the corporate headquarters is updating global policies or objects. Conflicts between simultaneous users are resolved through locking mechanisms in the database engine by the Management Server.

A single GUI can be used to control all the firewalls and clusters under the same management system. For example, the same updated security policy can be installed on all the firewall systems in the corporate network at the same time.

The main view of the GUI is called the *StoneGate Control Panel*. It contains a special toolbar *Launchpad*, from which the various manager applications can be started. Each manager is an application component used to perform a particular set of related tasks. For example, the Security Policy Manager is used to create, modify and delete security policies or rule bases; while the Network Element Manager enables the creation, modification and deletion of various network elements, such as hosts, routers, network links, and firewall clusters.

ILLUSTRATION 2.1 *The StoneGate Control Panel*



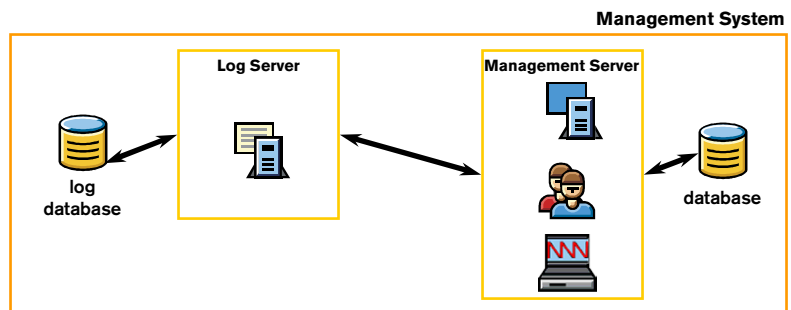
In addition to the StoneGate Launchpad toolbar, the StoneGate Control Panel presents the administrator with statistical information and graphical representations on the current state of the firewall clusters and engines. This information is processed by the monitoring system of the Management Server, as described below.

For detailed information about the GUI, please see Chapter 3, *GUI Overview and Implementation Design*, on page 53.

Management System

The management system is composed of the machine or set of machines that run the various management components. As illustrated in Figure 2.2 on page 27, the management system consists of four interrelated components: the Management Server, the Log Server, the user directory service, and the object databases. The Management Server and Log Server components interpret commands and data from the GUI and communicate that information to the firewall engines. In turn, data from the firewall engines is filtered through the management system and sent to the GUI for display.

FIGURE 2.2 Components of the management system



Management Server



The Management Server is the core server in the management system, and the central point of administration. A single Management Server can be used to manage an entire enterprise network of StoneGate clusters and single firewalls. It is in charge of maintaining policies, network elements, and communicating the administrator's instructions to the firewall engines.

In addition to the maintenance tasks described above, the Management Server also features a monitoring system. The monitoring system receives performance information and current node conditions from the firewall nodes. It processes the raw performance data and presents it to the GUI, which can then display the information to the administrator in a clear, comprehensive format. Illustration 2.2 shows how performance information is displayed in the GUI. Information is queried on demand and cached for a period of time, at the end of which it is discarded.

Additionally, the monitoring system provides several control commands for the firewall nodes. Through the GUI, the administrator can command a node or set of nodes to go online or offline or to reboot, for example. These control requests are passed through the monitoring system to the Management Server, which then communicates the requests to the firewall engines.

ILLUSTRATION 2.2 *Statistical data from the Management Server*



Log Server



Log Servers manage the log data generated by each firewall node. They store event information in a database, which can be queried, sorted, and exported. What data is viewed and how it is presented can be controlled through user-defined filters, which can also be used to filter data for export. Illustration 2.3 shows an example of log data that has been processed by the Log Server.

ILLUSTRATION 2.3 *Sample log data*

The screenshot shows the 'NetBIOS - Log Browser' window. It has a menu bar (File, Edit, View, Options, Help) and a toolbar with buttons for 'Current', 'Resume', and 'Pause'. Below the toolbar is a table of log data. The table has columns: Time, Originator, Facility, Type, Event, Action, Protocol, Src Addr, Dst Addr, Src Port, and Dst Port. The data rows show various network events, including connections and packet filters, with alternating green and orange background colors for rows.

Time	Originator	Facility	Type	Event	Action	Protocol	Src Addr	Dst Addr	Src Port	Dst Port
18.7.2002 10:41:42	Ganymede	Packet filter	Notification	Connect...		UDP (17)	192.168.3.101	10.0.0.10	1348	514
18.7.2002 10:44:20	Ganymede	Packet filter	Notification	New conn...	Allow	UDP (17)	192.168.3.101	10.0.0.10	1104	514
18.7.2002 10:45:48	Ganymede	Packet filter	Notification	Connect...		UDP (17)	192.168.3.101	10.0.0.10	1104	514
18.7.2002 10:49:36	Ganymede	Packet filter	Notification	New conn...	Allow	UDP (17)	192.168.3.101	10.0.0.10	1104	514
18.7.2002 10:50:30	Ganymede	Packet filter	Notification	Connect...		UDP (17)	192.168.3.101	10.0.0.10	1104	514
18.7.2002 10:54:35	Ganymede	Packet filter	Notification	New conn...	Allow	UDP (17)	192.168.3.101	10.0.0.10	1104	514
18.7.2002 10:55:49	Europe	Packet filter	Notification	Connect...	Discard	TCP (6)	192.168.3.101	212.20.3.1	1169	80
18.7.2002 10:55:53	Europe	Packet filter	Notification	Connect...	Discard	TCP (6)	192.168.3.101	212.20.3.1	1169	80
18.7.2002 10:55:58	Ganymede	Packet filter	Notification	Connect...		UDP (17)	192.168.3.101	10.0.0.10	1104	514
18.7.2002 10:55:59	Europe	Packet filter	Notification	Connect...	Discard	TCP (6)	192.168.3.101	212.20.3.1	1169	80
18.7.2002 10:56:12	Europe	Packet filter	Notification	Connect...	Discard	TCP (6)	192.168.3.101	212.20.3.1	1169	80
18.7.2002 10:56:38	Europe	Packet filter	Notification	Connect...	Discard	TCP (6)	192.168.3.101	212.20.3.1	1170	21
18.7.2002 10:56:41	Europe	Packet filter	Notification	Connect...	Discard	TCP (6)	192.168.3.101	212.20.3.1	1170	21
18.7.2002 10:56:48	Europe	Packet filter	Notification	Connect...	Discard	TCP (6)	192.168.3.101	212.20.3.1	1170	21
18.7.2002 10:57:01	Europe	Packet filter	Notification	Connect...	Discard	TCP (6)	192.168.3.101	212.20.3.1	1170	21
18.7.2002 10:57:22	Ganymede	Packet filter	Notification	New conn...	Allow	UDP (17)	192.168.3.101	10.0.0.10	1104	514
18.7.2002 10:58:01	Ganymede	Packet filter	Notification	New conn...	Allow	TCP (6)	192.168.3.101	212.20.3.254	1177	23
18.7.2002 10:58:02	Ganymede	Packet filter	Notification	New conn...	Allow	TCP (6)	192.168.3.101	212.20.3.254	1177	23
18.7.2002 10:58:02	Ganymede	Packet filter	Notification	Incomplet...		TCP (6)	192.168.3.101	212.20.3.254	1177	23
18.7.2002 10:58:02	Ganymede	Packet filter	Notification	New conn...	Allow	TCP (6)	192.168.3.101	212.20.3.254	1177	23
18.7.2002 10:58:02	Ganymede	Packet filter	Notification	Incomplet...		TCP (6)	192.168.3.101	212.20.3.254	1177	23
18.7.2002 10:58:03	Ganymede	Packet filter	Notification	New conn...	Allow	TCP (6)	192.168.3.101	212.20.3.254	1177	23

At the bottom of the window, it says 'LogServer 192.168.3.101, Local Time Ready' and '192.168.3.101 root Superuser Alert'.

Multiple Log Servers can be deployed in a StoneGate environment, and each firewall or firewall cluster can be assigned a different Log Server to improve performance and availability. In geographically distributed systems, this option is particularly useful. Individual nodes within a single cluster must log to the same Log Server, however. In the event that a node cannot communicate with its Log Server, the node can store information locally until communication is restored.

Database Engines



The Management Server stores the properties of network elements, rules, aliases, templates, VPNs, users, services, and other information in an internal database engine. This embedded database engine

provides StoneGate with a fast and robust SQL relational database. This self-maintaining database is installed automatically as a component of the Management Server.

The Log Server also uses an embedded database engine. This database allows the Log Server to store, query and export log data very quickly, using the same kind of a fast, robust SQL relational database. As with the Management Server's embedded database engine, this self-maintaining database is installed automatically as a component of the Log Server.

User Directory Service



User information is stored in the Management Server's internal database. The user directory information, although stored in the embedded SQL database, can be accessed through an LDAP API. This interface, or front-end, enables corporations with existing LDAP directory services to use those instead of, or in addition to, the built-in user directory.

Firewall Engines



The firewall engines are at the core of StoneGate and run on hardened Linux based operating system which is installed with the engine install. These are the servers that run the firewall itself. Each node has multiple network interfaces, and runs the engine to inspect and filter packets addressed to or sent from the network they protect.

Supporting systems, such as the configurable test subsystem, are also run on the firewall nodes. The kernel, protocol agents, firewall, test subsystem, and logging services each function independently. This independence makes the engine robust; an error or corruption in one service will not affect other services. The system can continue until repairs or regular maintenance corrects them. The engine on each node communicates with the management system by sending

performance statistics, state information, and log data, and by receiving policy updates and configuration changes.

The engine is configured through configuration files, each of which is loaded onto the nodes by the management system. These files are encrypted on the disk for each node, so that unauthorized access to the firewall itself will not reveal further information about the network. The configuration files are generated by the management system, based on input from the administrator through the GUI. Therefore, no manual reconfiguration of these files is needed.

Conveniently, the software on the nodes can be upgraded remotely from the GUI. Hence there is no need to physically access the firewalls to upgrade them to a new version.

All firewall engines, management system, and Log Servers authenticate communications with each other using public key cryptography and digital certificates with the help of Secure Sockets Layer (SSL) protocol. This ensures that only authorized nodes and an authorized management system can take part in the cluster's operation and configuration. The keys and certificates are automatically generated during installation. Each engine defined in a cluster also communicates with every other engine in that cluster, sharing information about current connections, load, and state information through Ethernet (IEEE 802) multicast.



.....
Tip: For more information about multicast, please see Appendix B, "Multicasting".
.....

In addition to the firewall engine, the firewall nodes each run a test subsystem. This robust, customizable system enables StoneGate administrators to define or enable additional tests to monitor and

respond to conditions on the firewall nodes. For example, if a network interface fails, the node can switch itself offline, have the other nodes redistribute the traffic to remaining online systems, and send an administrative alert.

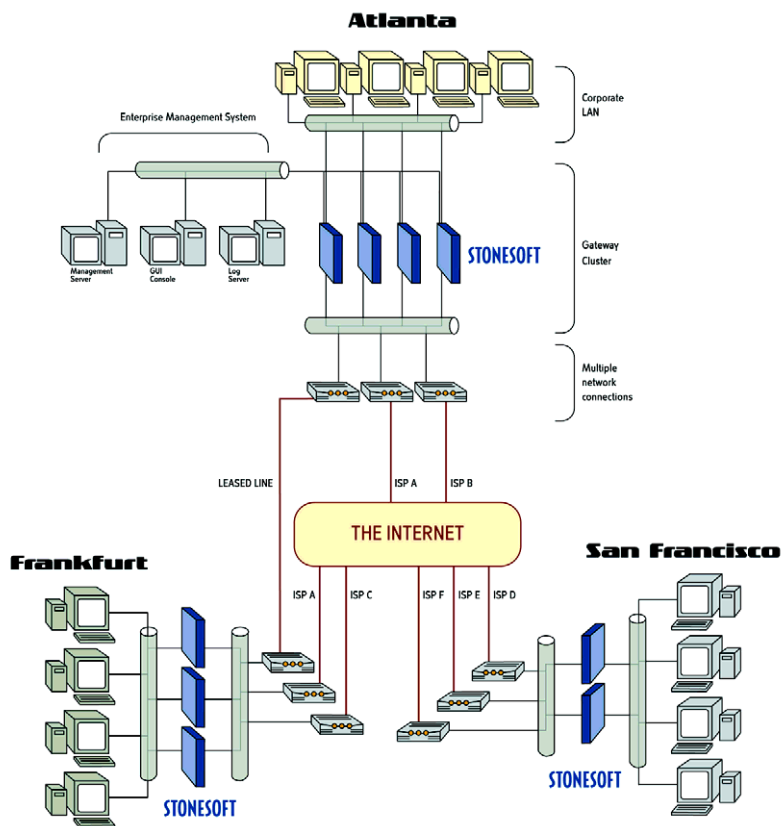
Architecture Benefits

Distributed Management

With this distributed architecture, the StoneGate administrator can use a single management system to monitor, control and manage multiple StoneGate firewall clusters throughout the enterprise network. For example, Figure 2.3 illustrates how an administrator can deploy a management system on a dedicated, trusted LAN segment at the corporate headquarters in Atlanta. The Log Server and Management Server can be deployed on a single server, or on multiple machines. An additional GUI client can be added, or the GUI can be run from a trusted system elsewhere.

With this management system, the administrator can define corporate-wide security policies, and describe the networks of remote offices in Frankfurt and San Francisco, for example. Firewall clusters in Atlanta, Frankfurt and San Francisco can all be monitored, controlled and managed from the same graphical interface interacting with this single, distributed management system.

The local administrators can be defined with different administrator rights to delegate the management tasks. They can be given certain permissions relating, e.g., to the maintenance and monitoring of the system or rights to configure specific types of elements. Please see section “*Administrator Levels*” on page 49 for further information on administrator management.

FIGURE 2.3 *Management of a distributed enterprise network*

Virtual LANs

A *Virtual Local Area Network* (VLAN) is a logical grouping of hosts and network devices appearing as a single Local Area Network (LAN) segment, regardless of its physical topology. This allows workstations and servers in the same VLAN to operate as if they were connected to the same physical network segment, thus forming a single OSI Layer 2 broadcast domain. The separation into different VLANs is implemented by *VLAN tagging* as defined by the standard IEEE 802.1q that is widely supported by various switches and operating systems. Network ports on a switch can be configured to make a

distinction between different VLANs by adding a two-byte VLAN tag on each Ethernet frame they pass between ports.

Control over network traffic is enhanced by limiting broadcast and multicast traffic within the VLAN. Thus, unnecessary traffic to other LAN segments is reduced. With the separation of the logical and physical LAN structure, the topology of a VLAN can follow for example a company's organizational structure by connecting the users of a department to the same VLAN, even if they are located in different physical LANs. This also reduces the cabling required as the separation to different segments is done by switches.

StoneGate supports virtual LAN (VLAN) tagging. It is able to process these tagged packets and thereby distinguish traffic belonging to different VLANs. When using StoneGate in connection to VLAN enabled switches, the different network ports of StoneGate can be configured for appropriate VLANs in the GUI. The benefit of using VLANs is that the number of physical interfaces needed can be reduced. In addition, it makes easier to deploy geographically distributed firewall clusters. Less physical interfaces means that less cabling is needed, say, between different buildings. It also reduces hardware costs. A single Ethernet interface can support up to 4094 VLANs.

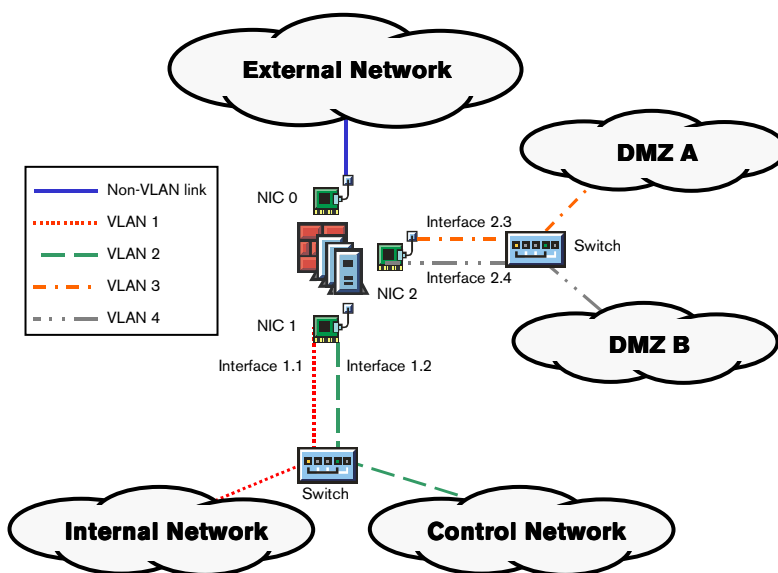
Figure 2.4 presents a simple example how VLANs can be configured with StoneGate. The firewall nodes have three network interface cards (NIC), but five isolated networks are required: external network connected to the Internet, DMZ perimeter network for the public Web and FTP servers, control network for managing the firewalls, and internal networks for the departments A and B.

In this case, the NIC 1 is configured as two logical VLAN interfaces: interface 1.0 for VLAN 0 and interface 1.1 for VLAN 1. The VLANs are configured to the switches to isolate the control and internal

network. NIC 2 is configured as interface 2.2 and 2.3 for VLANs 2 and 3, respectively.

If the implementation of the VLANs in the switches is secure, the switches are configured properly and the access to the switches is restricted, the traffic belonging to the different VLANs is securely isolated.

FIGURE 2.4 *StoneGate with VLAN tagging*



High Availability Technologies

In the past, firewalls could be made highly available through add-on software or redundant hardware solutions. Often, however, this would result in the transfer of a single point of failure to another network component—typically the network link. Other network services, such as Web servers, FTP servers, or content inspection servers would

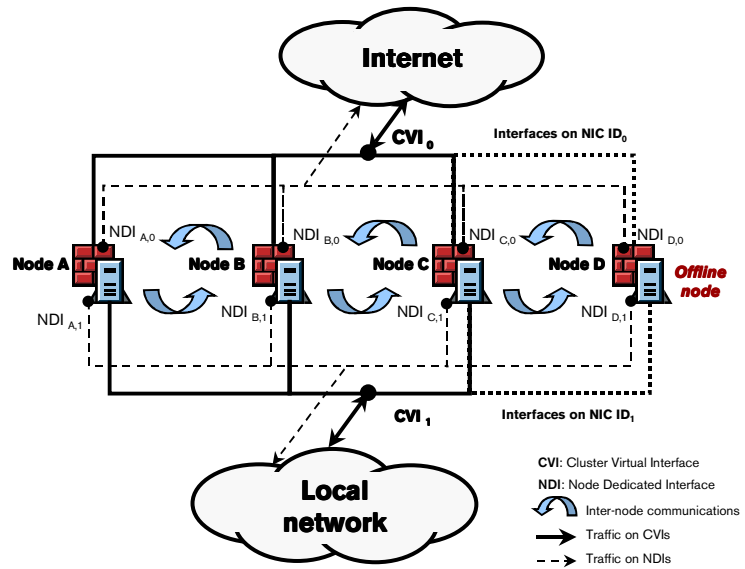
require yet more hardware and software to eliminate single points of failure in the enterprise.

The StoneGate High Availability Firewall includes several clustering technologies to reduce the complexity of achieving high availability. StoneGate's innovative clustering and load-balancing features provide several benefits over traditional solutions. In this section, we'll look at the different elements that can be clustered, and the different load balancing technologies that can be deployed with StoneGate.

Built-In High Availability

Traditionally, in order to achieve high availability on the firewall itself, it was necessary to add and maintain additional hardware switches, software clustering products, or special load balancing devices. With StoneGate, however, the clustering of the firewall engines is built into the product. Thus, StoneGate introduces true *built-in high availability*. The firewall engines dynamically load balance individual connections between the nodes in a cluster, transparently moving them to available nodes if a node becomes overloaded or experiences a failure. Using virtual IP addresses combined with MAC addresses, the firewall engines are seen by the rest of the network devices as a single entity, reducing the need for configuration changes elsewhere on the network. Figure 2.5 on page 37 illustrates a clustered set of firewall engines.

FIGURE 2.5 A firewall cluster



Each engine communicates with other engines in the cluster with a special *heartbeat* protocol, run on a dedicated network. Using a dedicated network ensures that the critical inter-node communications can take place, even in the event of a denial-of-service attack or periods of peak network traffic. Additionally, the dedicated network can be used to enable communication of connection tracking information and control data.

The cluster also provides for scalability, and maintenance during business hours. Individual engines can be added to the cluster at any time, scaling up to 16 nodes. Individual nodes can be taken offline during business hours and maintenance performed; connections that were using that engine will be transparently redistributed to other online nodes.

High Availability Connections

In addition to the distributed architecture of StoneGate's components, and the ability to cluster firewall nodes, StoneGate provides options for high availability of related network components. Internet Service Providers (ISPs) and virtual private networks (VPNs) have often been a single point of failure for enterprise communications. High availability was attainable, but at a cost of complexity. StoneGate eliminates these issues by introducing innovative technologies.

Multi-Link technology

As the role of Internet-driven business grows, the reliability of connections and constant availability of services is an absolute necessity for corporations. Even with the use of clustering products on servers, the corporate network would be subject to outages should the network link fail. In order to eliminate this single point of failure as well, it has also been necessary to deploy a battery of redundant external routers and switches. Often this would also require the use of complex routing protocols, such as *Border Gateway Protocol* (BGP) and *Hot Standby Routing Protocol* (HSRP), and peering arrangements through the ISPs providing the Internet connections. Implementing this type of solution results in high additional expenses including: redundant hardware, more expensive routers, additional ISP arrangement costs, etc. The network administrator would be faced with the daunting task of configuring and maintaining this complex network to achieve high availability.

Highly available network links

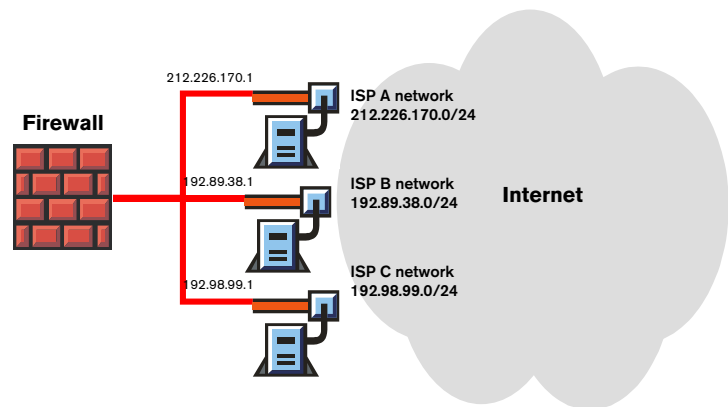
Stonesoft has a strong track record of producing software solutions for clustering and high availability. StoneGate was designed with high availability and fault tolerance in mind from the beginning. With StoneGate, a network administrator can now configure several ISPs with *Multi-Link technology*, which removes the problems related to the use of routing protocols. The resulting high availability of network links also reduces the need to use costly leased lines. Any IP-based

link with a dedicated IP address range can be used as part of a Multi-Link configuration;

- standard Internet connections
- ISDN connections
- leased lines.

As seen in Figure 2.6, an administrator can install multiple routers with links to different network providers. StoneGate can then be easily configured to load balance connections between the network links. If one of the routers or its link to the network provider should fail, StoneGate enables the end user to immediately re-establish connections through the remaining available links and routers. Note that even a single StoneGate firewall can be configured with a virtually unlimited number of ISP links.

FIGURE 2.6 *StoneGate Multi-Link technology*



Load-balanced routing

As an additional benefit, traffic can be load balanced across all available links, improving the overall throughput of the network. For example, using Stonesoft's patented *load-balanced routing* technology, users attempting to access Web sites will have their connection routed

through the fastest available link. The dynamic routing engine, which performs load-balanced routing, will send an initial packet out through each network link, and measure the response time. Based on the measurement, the connection will be routed automatically by StoneGate through the link with the best performance.

Increasing the bandwidth

Bandwidth requirements of corporations change, and traditionally, the only solution to get more bandwidth has been to pay more to the existing provider. Changing the ISP would result in the expensive change of public IP addresses. With StoneGate, you can increase the bandwidth by simply adding new ISPs to your network. StoneGate is able to operate the multiple ISPs as a single, virtual Internet connection, always providing the optimal physical link.

Benefits of Multi-Link technology

By implementing traffic management with StoneGate's Multi-Link technology, several advantages are realized:

- the networks are more secure, as there are fewer possibilities for traffic to bypass the firewalls through other network devices
- the networks are less complicated, since the firewall takes care of the load balancing, eliminating the use of special load redirectors, switches or other devices, which also may be single points of failure
- the cost and workload of maintaining and configuring the network are reduced
- scaling by adding new ISP connections is radically simplified
- the use of multiple ISPs increases the availability and bandwidth at use for critical Web applications and transactions
- load-balanced routing always selects the optimal route to a destination and thereby enhances performance
- high availability and load balancing are introduced to VPN traffic.

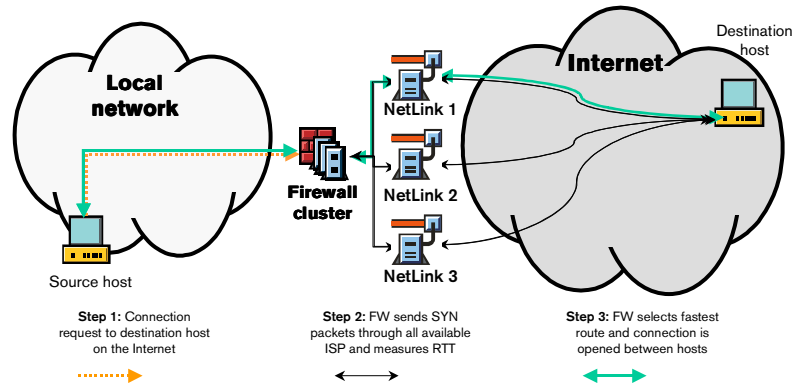
Multi-Link Technology Applied

In StoneGate, Multi-Link technology is applied in three unique scenarios. These are: outbound traffic management, Multi-Link VPNs, and inbound traffic management. Each scenario handles high-availability and load balancing issues in a different way. The following sections discuss each scenario briefly. A more detailed discussion regarding each scenario is held in the *StoneGate Advanced Implementations and Beyond* course.

Outbound traffic management

Outbound traffic management in StoneGate provides both high availability and load balancing between multiple network links (NetLinks). Load balancing between ISPs increases the overall throughput of your network, as it becomes easier to avoid congestion situations. For each new connection request, the StoneGate firewall can select from among the available links the fastest route for the connection.

Figure 2.7 illustrates the process for outbound traffic management decisions, using the round trip time of the first SYN packet as the basis for the load-balanced routing decision.

FIGURE 2.7 Outbound load balancing

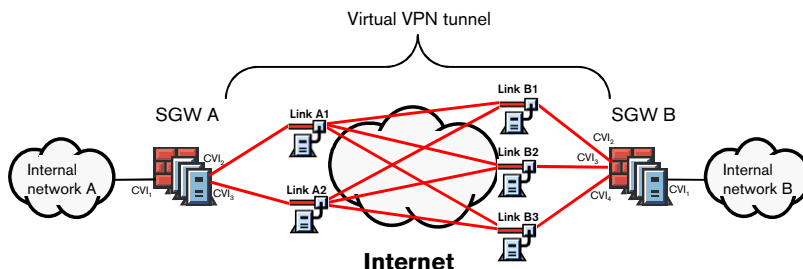
Multi-Link VPNs

VPNs are an increasingly critical component of enterprise networks. They enable trusted communication with remote offices or users through the insecure public Internet by creating encrypted “tunnels” to the corporation’s networks. Historically, VPN tunnels were subject to failure if the router or link with the network provider went down on either end. Often, only one VPN tunnel could be established with a remote site, placing bandwidth restrictions on an already limited connection.

With StoneGate, clustered, load-balanced IPsec VPN tunnels are now possible. A logical tunnel is created, consisting of one or more physical VPN tunnels through different ISPs. Figure 2.8 demonstrates a VPN consisting of two physical links at one end and three at the other. In this example configuration, StoneGate can use six different *subtunnels* forming one virtual VPN tunnel to load balance VPN traffic between the sites. Therefore, if one subtunnel collapses because a link to the network provider on either end fails, the connections using that tunnel will be transparently shifted to another subtunnel established

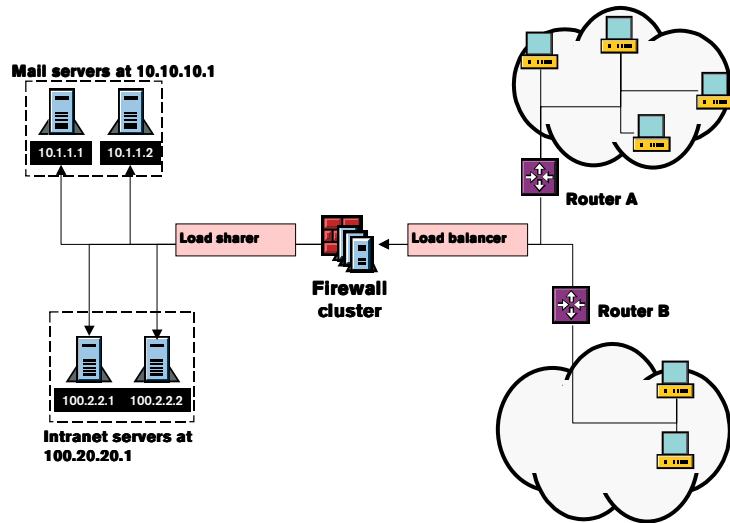
with that remote site. Should all ISP links be down, StoneGate can take standby links, such as leased lines or dialup connections, into use. This way, StoneGate introduces true fault tolerance and high availability to VPN traffic.

FIGURE 2.8 *Clustered VPN tunnels*

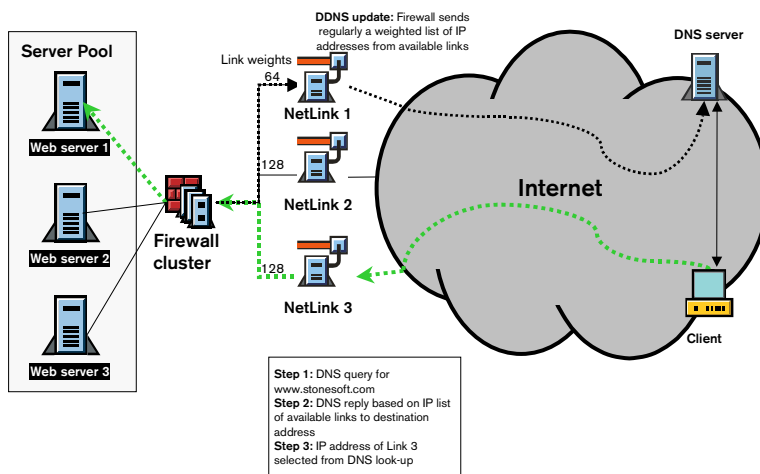


Inbound traffic management

In addition to maintaining links to the Internet, StoneGate can load balance connections to different network services provided by the company. Take Web servers for instance. With existing technologies, it was often necessary to deploy additional load balancing switches, clustering software or special load balancing hardware to ensure high availability and improved performance. Once again the network administrator was faced with additional complexity, as seen in Figure 2.9, to achieve high availability on the network.

FIGURE 2.9 *Traditional inbound load balancing*

With StoneGate, the administrator can configure load balancing to network services taking advantage of the Multi-Link technology and server clustering. For example, a Web server pool can be set up on a DMZ network, as illustrated in Figure 2.10. StoneGate can then load share connections to the pool of Web servers with load balancing between ISPs. The selection of the ISP is based on dynamic DNS updates. StoneGate also constantly monitors the availability of servers and ensures that connections from a particular source address are handled by one server only.

FIGURE 2.10 *StoneGate inbound traffic management*

As previously explained, each of these features will be covered in more detail in the *StoneGate Advanced Implementation and Beyond* course. With this overview in mind, however, this unit will now present the design of the firewall engine in further detail.

Multi-Layer Inspection

The firewall engine of StoneGate incorporates the best principles of firewall architectures with its unique *Multi-Layer Inspection* technology. The StoneGate firewall engine is run on servers, referred to as firewall nodes, and it is responsible for the enforcement of the corporate security policies. It inspects packets arriving at each interface and determines whether they are permitted to cross the gateway to the next network, based on rules defined by the network administrator.

A StoneGate firewall is capable of inspecting packets and making traffic handling decisions at different layers. With Multi-Layer Inspection, StoneGate is able to keep temporary records of accepted connections and their state in the appropriate context. This improves the overall performance of the firewall also, since by recording the state of connections, it can handle entire connections instead of individual packets. StoneGate stores the state information of each connection and checks that any packet arriving at the firewall is either an opening packet or belongs to an already accepted connection. When packets arrive, StoneGate checks them first against active, legitimate connections before proceeding to the rule base checking. If a connection has a registered state, the packets following the valid opening packet don't need to be checked against the rule base. This speeds up the firewall performance considerably without compromising security. Packets belonging to an accepted connection are allowed to pass after some manipulation, such as address translation, load balancing, decryption, defragmentation, or application-level processing.

A crucial aspect of StoneGate's Multi-Layer Inspection is represented by *Protocol Agents*, which function as extensions to the core operation of the firewall engine. While the engine itself takes care of basic connection layer (TCP, UDP, and ICMP) handling, the different Protocol Agents handle any related application-level information. They validate application layer data and, if required, modify it (relating to network address translation). They also open related connections whenever required. They handle data in connections—and not in individual packets—and on the basis of this data, allow other related connections. Tasks that they perform include checking FTP control connections and redirecting traffic to content inspection servers. The use of more complicated protocols, such as FTP or H.323, requires the use of protocol agents for better security. Protocol agents can be associated, for example, with FTP, HTTP, SSH, Remote Shell, and Oracle Net8 protocols. Protocol agents are covered in detail in the *StoneGate Advanced Implementation and Beyond* course.

Next, we shall take a closer look at how the checking against the rule base is carried out.

How StoneGate Examines Packets

The header on each packet arriving on an interface is examined for the source and destination IP address and ports. Other matching parts of rules include authentication and validity time. Certain rules may be applied only if a user is successfully authenticated by the method defined in the rule. Other rules may only apply on certain times of day, e.g., during office hours.

In case of a StoneGate cluster, the firewall's load balancing filter determines which engine in the cluster will actually process a packet, while all other engines immediately discard it. In case of a single firewall, there is no load balancing as there's only one node to handle the traffic. The engine that will handle the connection checks the current connection tracking information to see if the packet is part of an established connection. If it is, there is no need to further apply access rules, and the packet can move on to NAT rules and traffic management. If the packet is not part of an existing connection, the packet is compared with the rules in the installed rule base (including possible subrule bases) until a match is found. If there is no match, it is discarded. If a match is found the appropriate action is taken, as set in the rule. If the action is to *allow* the packet, NAT rules are applied next. Log settings and other options are then taken into account, a routing decision is made, and the packet is let through the firewall.

If a rule specifies that a matching packet must be *discarded*, the rule base traversal is immediately stopped and the packet is silently dropped. The fate of the *refused* packets is the same except that an ICMP error message is delivered to the sender to inform that the packet was not allowed.

If the packet matches a rule with *Continue* defined as the action, the rule options of that rule are written in the memory but the matching

process continues until a rule that determines the way the packet is handled is found.

A rule with *Jump* as the action means that a separate subrule base is applied for any matching packets. This speeds up the matching process.

The VPN traffic is handled a bit differently than unencrypted connections. If the action is *Apply VPN*, a packet is let through if it belongs to the specified VPN. If it belongs to another VPN, the matching process continues. The *Enforce VPN* action, on the other hand, is more strict and if the packet matching a VPN rule doesn't belong to the specified VPN it is dropped immediately.



.....
Note: StoneGate operates under the security principle of "that which is not expressly permitted is denied."
.....

StoneGate Administration

StoneGate also offers a flexible system of access control for configuring and maintaining the firewall clusters, networks, users and other related components. Access to the GUI is provided to three different user levels:

- Superuser
- Editor
- Operator

These levels enable the delegation of administrator tasks in enterprise networks. Additional privileges can be granted to lower levels. The network administrator can create multiple accounts of each type, with unique user IDs and passwords.

Administrator Levels

Let's examine each of the administration levels a bit further. This section will provide an overview of access control and administrator accounts. For a more detailed explanation of administrator levels, please see Chapter 7, *Administrator Management*, on page 105.

Superuser

The Superuser is the highest level of access, much like the Administrator account in Windows NT or root in Unix. This level enables the creation of other administrator accounts, as well as the ability to create, modify or delete network objects, rule bases, templates, users and so on. The Superuser can also grant additional access privileges to Editors or Operators.

Editor

This level is the intermediate access level. Editors can modify some objects, users, and rule bases—but by default they cannot modify templates, create or modify editors, or delete objects. Additional privileges can be given to them by Superusers.

Operator

These are the least-privileged accounts, enabling the monitoring of systems, viewing of rule bases and configuration information. However, Operators cannot modify elements by default. They can be granted additional privileges by a Superuser, or by an Editor who has been given that privilege by the Superuser.

Summary

In this unit, you have explored the distributed architecture of StoneGate firewalls. You should now be familiar with major components, such as the GUI, management system, and firewall

engines, and have a better understanding of the individual components of the management system.

You should have also explored various new load balancing and clustering solutions introduced in StoneGate, including Multi-Link technology, load-balanced routing technology, and clustered VPNs.

You should be able to explain the process of packet inspection by the firewall engine, and its configuration. And finally, you should have an overview of the three main types of administrator accounts available in StoneGate.

Review Questions

- What are the three main components of StoneGate?
- What is the difference between “Management Server” and “management system?”
- Describe how outbound load-balancing of StoneGate works.
- How is high availability of VPN traffic achieved with StoneGate?

GUI Overview and Implementation Design

Buying the right computer and getting it to work properly is no more complicated than building a nuclear reactor from wristwatch parts in a darkened room using only your teeth.

- Dave Barry

This unit presents two related areas of the StoneGate product. In the first part of this unit, you will be presented with a brief overview of the StoneGate graphical user interface, which provides the framework for network administrators to set up and fine-tune the StoneGate system. It follows with a brief overview of the tasks you must perform to install StoneGate.

The second part features an overview of the StoneGate implementation, including options that enable you to distribute components in different ways that suit the needs of your particular network.

Objectives

Upon completing this unit, you should be able to:

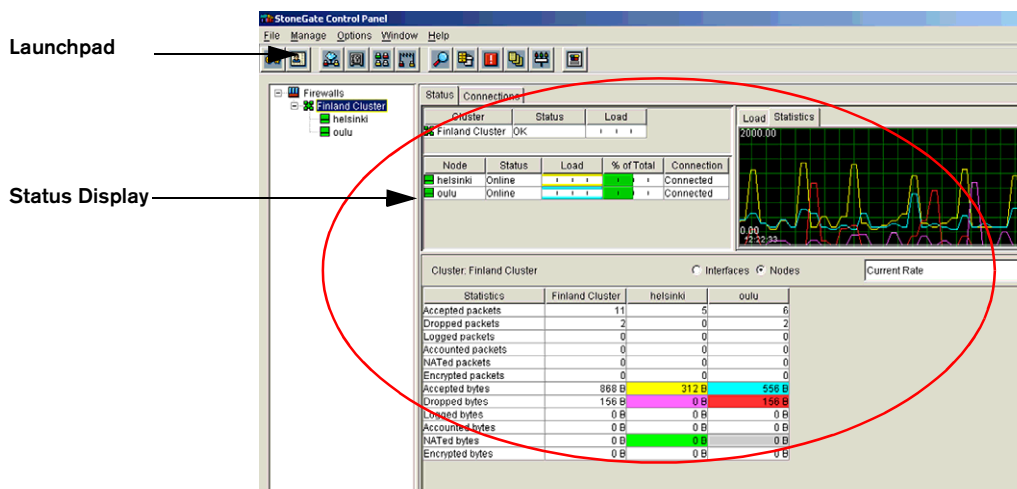
- identify the two main areas of the StoneGate Control Panel
- describe the role of each manager application available from the StoneGate Launchpad
- outline the basic tasks for installing StoneGate
- explain at least two ways the StoneGate components can be distributed in a corporate enterprise network.

StoneGate Graphical User Interface (GUI)

The StoneGate GUI, or Administration Client, can be installed on multiple systems, and is supported on Solaris[®], Linux[®], and Microsoft[®] Windows[®] platforms. Because StoneGate supports multiple administrator accounts, you must first login with a user ID and password to work in the GUI. When it is first launched, the Administration Client will present you with a login screen.

After you have successfully logged in, you will be presented with the *StoneGate Control Panel*. The StoneGate Control Panel can be divided up into three main areas: Tree View, Status Display and Launchpad.

ILLUSTRATION 3.1 *The StoneGate Control Panel*



Tree View

The *Tree View* in the leftmost panel displays all the firewall systems managed by the Administration Client. You can expand the tree to show all the clusters and all the nodes under each cluster. When you select one of the clusters or nodes the Status Display will present you with statistical data on that component. You can also select several

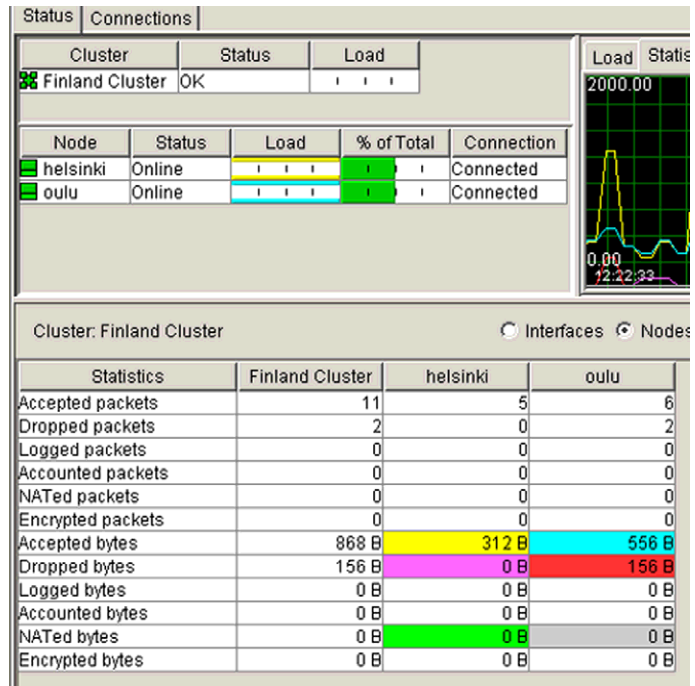
control commands by right-clicking on the selected cluster or node. If you want, you can choose not to show monitoring data on a given firewall system by selecting the **Not Monitored** option.

Status Display

The *Status Display* is your main window for monitoring the operation of the firewalls. Here you can see at-a-glance the operational status of each firewall engine. The status display presents you a wide variety of real-time data from the monitoring system of the Management Server. You can select the firewall clusters or individual nodes for which statistical information is displayed.

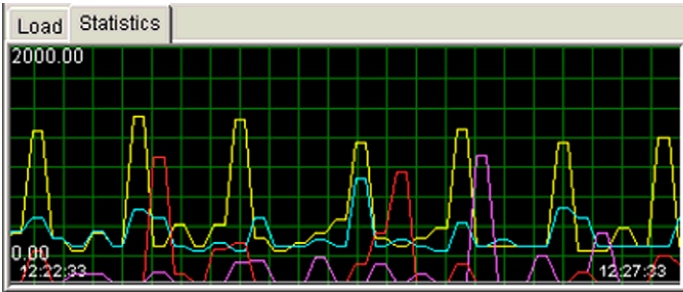
Status information is shown for the whole firewall (cluster or single). This status information depends on whether the firewall or nodes are still being installed and configured or whether they are already up and running.

The statistics tables keep track of the number of packets and bytes that are allowed, discarded, logged, accounted, NATed, and encrypted by the firewall. You can examine the statistical data on a per-node or per-interface level.

ILLUSTRATION 3.2 *Statistics tables*

In the area on the right side of the monitoring view, a real-time line graph will display up to six different data points in different colors, selected from the statistics tables. The load average and the percentage of total load per node can be visualized here as well. In addition to the current rate, you can also choose to check the average rates for each statistical parameter for the last minute, last hour, and last 24 hours.

ILLUSTRATION 3.3 *Statistics—graphical view*



When you switch to the Connections tab, you can drill to comprehensive log data on the open connections. By applying a pre-defined filter to that data, you can have only the relevant log entries displayed. This way, it's easy for you to pick out specific types of connections from the bulk of log data.

ILLUSTRATION 3.4 *Current connections*

Status Connections										
No Filter	Edit Filter	<input checked="" type="checkbox"/> Resolve	Update Speed:	Normal	Terminate					
Time	Originator	Protocol	Src Addr	Dest Addr	Src Port	Dest Port	NAT Src Addr	NAT Dest Addr	NAT Src Port	NAT Dest Port
20.7.2001 10:46:06	Oulu	TCP (6)	Oulu	web 1	2582	SG-log				
20.7.2001 10:46:04	Oulu	TCP (6)	Oulu	web 1	2581	SG-log				
20.7.2001 10:46:04	Helsinki	TCP (6)	Helsinki	web 1	2558	SG-log				
20.7.2001 10:46:03	Helsinki	TCP (6)	Helsinki	web 1	2557	SG-log				
20.7.2001 10:46:03	Oulu	TCP (6)	Oulu	web 1	2580	SG-log				
20.7.2001 10:46:02	Oulu	TCP (6)	Oulu	web 1	2579	SG-log				
20.7.2001 10:46:01	Helsinki	TCP (6)	Helsinki	web 1	2556	SG-log				
20.7.2001 10:46:00	Helsinki	TCP (6)	Helsinki	web 1	2555	SG-log				
20.7.2001 10:45:50	Oulu	TCP (6)	Oulu	web 1	2570	SG-log				
20.7.2001 10:45:49	Oulu	TCP (6)	Oulu	web 1	2569	SG-log				
20.7.2001 10:45:49	Helsinki	TCP (6)	Helsinki	web 1	2546	SG-log				
20.7.2001 10:45:48	Oulu	TCP (6)	web 1	Oulu	1066	SG-monitor				
20.7.2001 10:45:47	Helsinki	TCP (6)	web 1	Helsinki	1067	SG-monitor				
20.7.2001 10:45:47	Helsinki	TCP (6)	Helsinki	web 1	2545	SG-log				
20.7.2001 10:45:44	Oulu	TCP (6)	web 1	Oulu	1066	SG-monitor				
20.7.2001 10:45:41	Helsinki	TCP (6)	web 1	Helsinki	1065	SG-monitor				
20.7.2001 10:45:04	Oulu	TCP (6)	web 1	Oulu	1054	Idap (Light...				
20.7.2001 10:45:03	Helsinki	TCP (6)	web 1	Helsinki	1055	Idap (Light...				
18.7.2001 11:34:48	Oulu	UDP (17)	Oulu	225.1.1.2	1030	SG-statelev...				

Launchpad

Above the status display area there is a row of buttons, called the *StoneGate Launchpad*. From the Launchpad, individual application components, or managers, can be run. With each of these managers you can easily perform all the tasks related to particular firewall

functions, such as creating and managing security policies, VPNs, services, or users.

ILLUSTRATION 3.5 *StoneGate Launchpad*



Administrator Manager



The Administrator Manager allows the Superuser of the system to create, modify, and delete administrator accounts. By creating administrator accounts the management of a distributed system can be delegated to a number of administrators with different permission levels.

User Manager



The User Manager is for creating, editing, and deleting regular user accounts. Unlike administrator accounts, user accounts created with the User Manager are used to authenticate with the firewall engines, so that, e.g., employees in a corporation can access restricted resources in the enterprise network. Different authentication methods can be required from different users.

Network Element Manager



With the Network Element Manager, you can create, modify, or delete the various objects, or elements, on your networks. These elements include, e.g., routers, hosts, networks, address ranges, and firewalls. The elements you define can then used when creating security policies and configuring routing and anti-spoofing.

Security Policy Manager



The Security Policy Manager is used to design and manage security policies. Security policies are defined by rules which are created in the

Access Rule Editor and the NAT Rule Editor windows of this application.

Services Manager



The Services Manager manages services and protocol agents with which you can control different types of traffic in the access rules. A service typically consists of one or more protocols and port destination fields. Services falling under the following protocol types are supported: TCP, UDP, ICMP, IPsec, IP-proto, and SUN-RPC.

VPN Manager



The VPN Manager enables you to define a global policy for encryption schemes and to establish encryption policies for specific types of traffic between sites. The configuring of VPNs with the manager is made easy and flexible. You can also monitor the status of the VPN tunnels.

Filtering Profile Manager



The Filtering Profile Manager creates and maintains log data filters used to filter out specific types of log data. This is useful, e.g., when you want to put focus on certain type of traffic. The manager can also be launched from the Log Browser application.

Log Data Manager



With the Log Data Manager, you can set up management tasks to be performed either manually as needed, or according to a definable schedule. For example, if the number of log records to be stored is constantly high, unnecessary data can be scheduled for regular deletion or archiving.

Alert Notification Manager



Alerts are managed with the Alert Notification Manager. Alerts can be customized by definition (i.e., the name of the alert or the text of the

alert message) and by the way the system notifies the administrator of the alert message.

Log Browser



The Log Browser shows log data that is stored in the database or being currently delivered to the Log server. You can also browse archived log files with the same application. To view specific log data, administrators can use filtering profiles (see *Filtering Profile Manager above*). Stored and archived log entries can be displayed either starting from, or ending at a given time.

Log Pruning Filter Manager



The Log Pruning Filter Manager is where you define how logs should be pruned using the profiles created in the Filtering Profile Manager. You can cope with capacity limitations by pruning log data, for example, when the same rule generates both useful and useless log entries.

Audit Manager



With the Audit Manager, the Superuser of the system can track which types of actions have been made on the management system or which system generated events have taken place. This is helpful, e.g., when needing to know what kind of configuration changes have been made and when. The audit data can also be archived.

License Manager



The License Manager displays the licensing status of the current StoneGate system, retrieves new license files from the license server, and uploads retrieved files in the runtime system.

Customizing the Display

Many aspects of the StoneGate display can be customized for your convenience. Toolbars can be detached, the panels resized or hidden, and the columns for different tables either hidden or shuffled into a different order. When you close the particular component you are working on, you will be prompted to save your current configuration so that your custom display will be always available.

Installation Overview

You should now have a familiarity with the user interface of StoneGate. Next you'll review the basic steps to the installation of StoneGate. StoneGate has a different approach to installation than traditional firewalls. Since everything is performed easily through the GUI, mediated and controlled by the management system, the order of installation is typically:

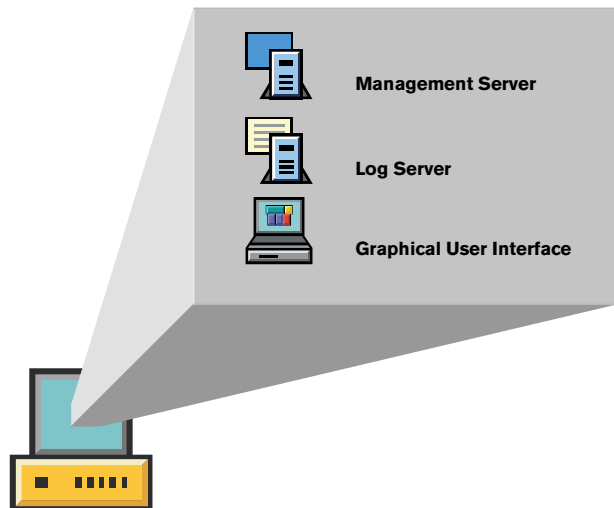
1. Install the management system and GUI client from the StoneGate CD. There are many ways of deploying these components, some of which are outlined in the next section, *"Implementation Strategies"*, below.
2. Install the licenses. You cannot proceed to work with StoneGate until valid licenses are installed. For more information about licenses, visit the Stonesoft License Center at <http://www.stonesoft.com/licenses/>.
3. Request the Log Server's certificate. This step also leads to the automatic creation of the Log Server element.
4. Create at least one firewall element in the GUI, defining all of its interfaces.
5. Save the initial configuration of the newly-created firewall element.
6. Install the firewall engine(s) from the CD and load the initial configuration on them.
7. Create your security policy, define routing, check anti-spoofing, etc. All of this is done in the GUI.

8. Install the security policy and your StoneGate firewall system is fully functional.

Implementation Strategies

From an implementation standpoint, one of the most useful StoneGate features is the ability to distribute components of the management system on different machines. In addition, each machine can also be running a different platform. Administrators can therefore run the Log Server on Linux, for example, while running the Management Server on the Solaris[®] operating system, and the user interface on the Microsoft[®] Windows[®] platform.

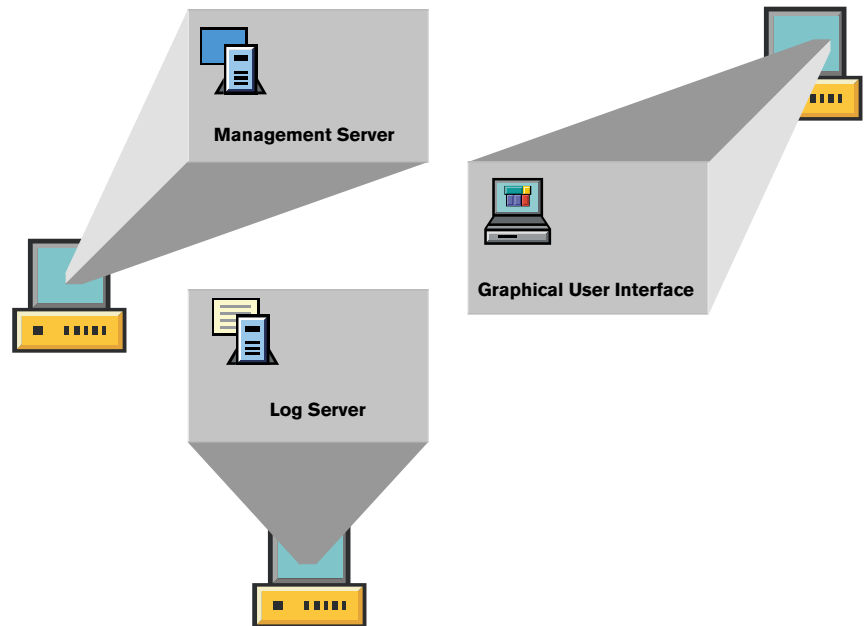
FIGURE 3.1 *Single management system*



For example, one possible implementation, illustrated in Figure 3.1, is to install the Management Server and the Log Server, plus a copy of the user interface, on the same machine. This strategy has multiple cost saving benefits. First, two servers on the same machine minimizes the amount of required hardware for the firewall implementation. This, of course, reduces the cost of the overall

implementation. In addition to reducing hardware costs, however, this also simplifies installation, thereby reducing administrative time and costs as well.

FIGURE 3.2 *Distributed management system*



Another possible implementation strategy, illustrated in Figure 3.2, is to install the Management Server, Log Server, and user interface on different machines. This strategy can enable better performance, since each machine is performing only a portion of the work required by the entire management system. This strategy also allows administrators more flexibility. For example, since each machine can run any platform, an administrator might choose to run the user interface on a Microsoft Windows NT[®] platform, so that other administrators have a familiar operating system. The Management Server may run on a Sun[®] Solaris[®] system, taking advantage of that dependable and powerful architecture. However, the Log Servers might be run from

Linux[®] machines to minimize the hardware requirements, while at the same time allowing for inexpensive, yet large, amounts of hard drive space for the log database. Since the entire design uses centralized administration through the GUI, the location of each of these servers is immaterial.

Summary

The StoneGate Control Panel gives you access to all the applications and monitoring information that is required for firewall administration. StoneGate GUI incorporates several applications—or managers—which all can be started handily from the same toolbar, called the Launchpad. Each of the managers takes care of a set of related tasks. The status display of the StoneGate Control Panel presents comprehensive statistical data from the nodes, both in table and graphical format. You can monitor all the clusters and nodes in the same window. You can also customize the Control Panel for your convenience. StoneGate management system components can be distributed on different machines. Moreover, they can be run on different platforms and can even be distributed to different locations.

Review Questions

- What are the two main areas of the StoneGate Control Panel?
- What are at least three managers available from the StoneGate Launchpad?
- What type of statistical data is available in the status display?
- What are the benefits of implementing the Management Server and the Log Server on separate machines? How about on the same machine?

Routing and Anti-Spoofing

Getting there is half the fun.

– Unknown

This unit covers how StoneGate incorporates routing configuration and anti-spoofing measures on each interface. StoneGate makes it unnecessary to configure basic anti-spoofing policies on the firewall engines; network elements in the GUI can be used to configure routes with simple drag-and-drop operation. In this chapter, we'll examine how StoneGate handles these tasks, what aspects are configured automatically, and when additional configuration information may be required.

Objectives

Upon completing this unit, you should be able to:

- explain basic firewall routing principles
- create basic default routes in StoneGate
- describe the additional routes created automatically
- explain IP spoofing attacks, and why firewalls can be vulnerable to them
- summarize StoneGate's main defenses against IP spoofing attacks
- understand policy routing and static IP multicast routing.

Firewall Routing

In addition to examining packets, a firewall has to determine how to properly route packets so that they can get to their destination. Traditionally, the only routes an operating system will configure automatically are those to directly-connected networks. That is, the system will take the IP addresses assigned to an interface, and use that address as the route to the network which includes that IP address. For a server to contact any other network, the machine must be given a default gateway, to which it will send any packets whose destination is not reachable locally.

For firewalls, routing has typically been more complex. In addition to having a default gateway for sending packets to other networks, administrators would have to configure static routes for other corporate networks. These static routes serve as directions for the server, instructing it which gateways to use to get packets to particular destinations. With firewall clusters, the correct routing information would have to be entered on each node individually, increasing the possibility that one or more nodes would have incorrect routing information.

Table 4.1 presents a hypothetical routing table that might be used on a firewall. In it, we see that the default gateway is set to 10.0.1.254. To reach that router and that network, the second route indicates that the ge0 interface should be used. The next one is to a 192.168 network, also directly connected—perhaps a DMZ network. This route is followed by the 172.16.4.0 network—perhaps an internal network. The 127.0.0.0 network is automatically created by most operating systems, and is for the local machine to talk to itself, commonly called the *loopback interface*. In order to get to a special network in Research and Development (192.168.2.0), the firewall is instructed to use 172.16.4.50. If the information for this route is entered incorrectly, say

172.16.4.5 is used as an address instead, then traffic would be unable to get to its destination.

TABLE 4.1 *A sample routing table for a firewall*

Destination	Gateway	Netmask	Flags	Metric	Ref	Use	Interface
0.0.0.0	10.0.1.254	0.0.0.0	UG	0	0		ge0
10.0.1.0	10.0.1.5	255.255.255.0	U	0	0		ge0
192.168.1.0	192.168.1.5	255.255.255.0	U	0	0		ge1
172.16.4.0	172.16.4.5	255.255.255.0	U	0	0		ge2
127.0.0.0	0.0.0.0	255.0.0.0	U	0	0		lo
192.168.2.0	172.16.4.50	255.255.255.0	U	1	0		ge2

Often, these complex tables and configurations would lead to errors, possibly impacting network performance, security, and availability. Network address translation (NAT) on some firewall products would require additional routing information to get packets to their proper destination. For example, consider a Web server using 192.168.4.20, and translated to an external address of 64.34.24.50. In order to handle this translation, the firewall would need an additional route, informing it that any traffic to 64.34.24.50 should be routed to 192.168.4.20. An enterprise with Web server pools, DNS servers, mail systems, FTP servers and such would require many additional routes to handle the additional address translations. To avoid some of this lengthy configuration one could choose to use routing protocols—enabling the systems to configure routes automatically. Unfortunately, most routing protocols have serious security flaws.

Routing Protocols

Routing protocols, such as *Router Information Protocol* (RIP), or *Open Shortest Path First* (OSPF) have security weaknesses that can be exploited by a hacker. RIP, for example, receives routing updates from other machines running RIP, and will adjust routes without requiring

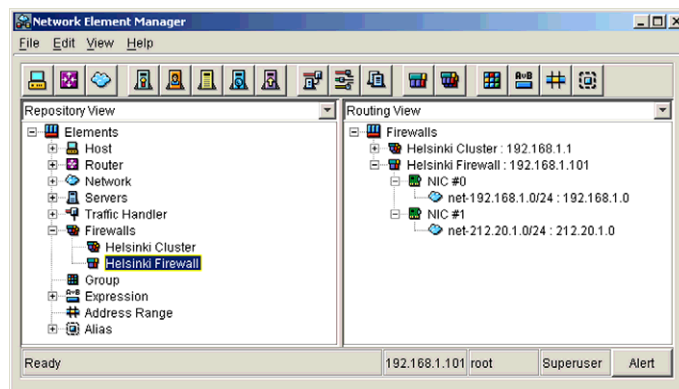
any authentication. OSPF uses MD5 digests and basic password authentication to secure the communications with other authorized routers, however the security is only minimal, and relies on good passwords.

If a hacker is able to make routing changes, they can redirect packets to rogue servers, capturing data as it flows by, for example. They can also perform a simple denial-of-service attack by convincing a gateway that its default route is to the internal network, preventing it from sending data out to the world.

StoneGate Routing

With StoneGate, the network administrator eliminates the need to run routing protocols. Instead of complex routing tables and individually configuring nodes, the administrator can create routing information through the Control Panel's *Routing View*. Routers can be added, and combined with automatically generated network elements, to the interfaces of the firewalls. Once created, this configuration is saved to all nodes in a firewall cluster at the same time, and handled by the Management Server, so the possibility of errors is minimized.

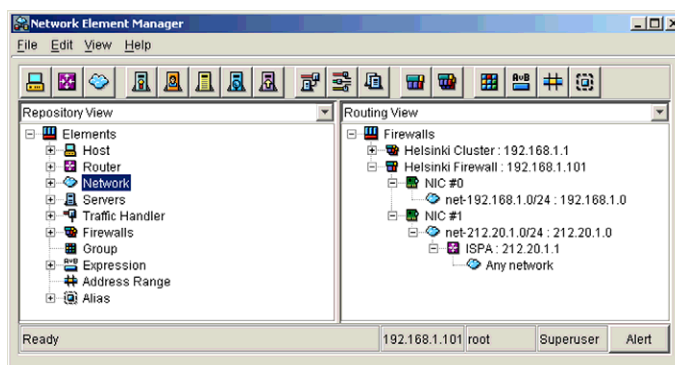
ILLUSTRATION 4.1 *An example of StoneGate's automatic network elements*



With the Network Element Manager, an administrator creates a firewall element (or a firewall cluster element), where the IP addresses of each interface are defined. When the Network Element Manager is run for the first time, a default network element, called *Any Network*, is created (see Illustration 4.2). Based on the IP address information for the firewall cluster, additional network elements are automatically generated by StoneGate for each network. Additional network elements can be created by the administrator as well. Once the networks are created, router elements can be added.

Illustration 4.2 shows the routing view of a complete network. In this example, StoneGate will know that the router **ISP A** is connected to the same 212.20.1.0/24 network to which StoneGate cluster **Finland** is connected. The firewall nodes will understand that **ISP A** is their gateway to any other network (that they don't otherwise have a route for), and will configure their routing tables automatically to reflect this design.

ILLUSTRATION 4.2 *Routing view of a network.*



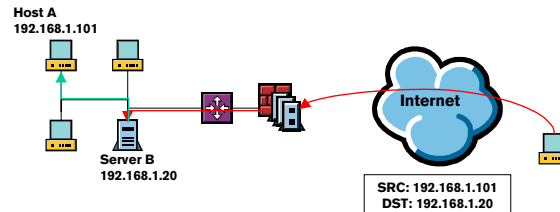
When a network element, such as a host or network is defined and cloaked behind NAT, routing information for those objects is generated based on the network information and translated addresses. Traditionally, an administrator would have to add a route from the outside address to the host's correct internal address, so that the

firewall could forward the packet to the proper destination. With StoneGate, such additional information must no longer be defined by the user. The firewall engines take care of all of the necessary routing changes each time a NAT rule is created.

Because the routing information is generated for all nodes in a cluster simultaneously, the administrator can configure, change, and view routing information through the GUI, without having to configure static routing tables on each firewall node or rely on insecure routing protocols to update information. With a distributed network environment, the management system can ensure the each node of each firewall cluster has the same, correct routing information that they require to function properly.

IP Spoofing

Another related problem with interfaces is a type of attack known as *IP spoofing*. IP address spoofing is the act of changing the source IP address in an IP packet in order to gain unauthorized access. With IP address spoofing, the firewall is convinced that packets are originating from an authorized source, when in fact they are coming from a compromised machine used by a hacker. One of the simplest IP spoofs is to send a packet with a legitimate internal address to a firewall's external interface. Since the operating system normally doesn't interpret packet direction, the packet is assumed to belong on the internal network, and so it is routed there accordingly. Figure 4.1 illustrates such an IP spoof.

FIGURE 4.1 *Basic source IP address spoof*

Another basic IP spoof is where an attacker on the Internet claims to be a host that is trusted by the firewall. If the firewall only relies on the available IP address information, the firewall will accept the connection. There are also more complicated IP spoofing attacks, but newer tools are making such attacks easier.

At first, it may seem as though the IP spoof is harmless as all responses will return to the legitimate host (Host A, in our example), instead of the attacker. However, there are three conditions where the hacker may not care:

- First, if the hacker is situated on the same network between the destination server (Server B), and the legitimate source (Host A), they can see the reply and carry out the conversation as if they were the real source.
- Or, the attacker may not even care what the reply is. In this type of attack, sometimes called a “blind spoof”, the attacker may be able to guess what the reply will be, or may simply attack to tie up a server with worthless communications (denial of service).
- Another option is for the attacker to intentionally not want the reply. They may intend the server and the legitimate host to eventually engage in conversations about why the Host A is

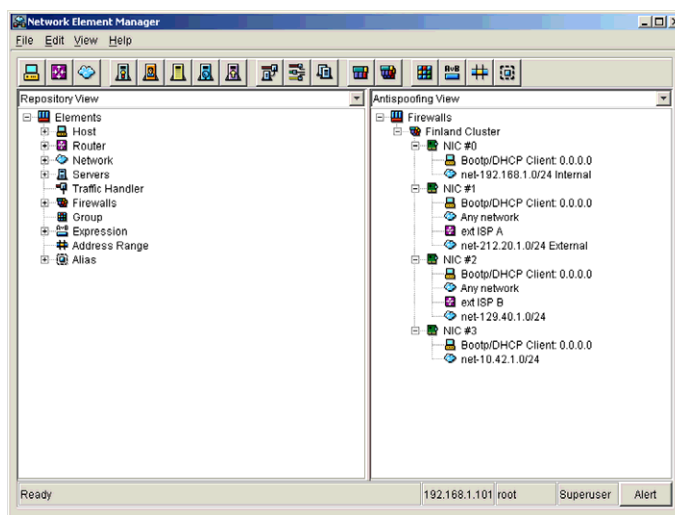
attacking Server B, and therefore deny legitimate services while they argue over which machine started the “fight”.

Since IP spoofing attacks have been a known exercise in hacking for several years now, many firewall vendors provide an option for configuring *anti-spoofing rules* for each interface. These rules essentially allow the administrator to instruct the firewall which source and destination addresses should be valid on any given interface. If a packet arrives on the external interface, for example, claiming to belong on the internal interface, the firewall would know to drop the packet.

But the traditional approach, and its methods for configuring such rules, must be understood and clearly defined by the network administrator. By default, the firewall won’t perform such a check on the direction of the packets. Since this oversight can lead to a potential security risk, StoneGate has been designed to automatically and intelligently determine the proper anti-spoofing measures to deploy.

Once a firewall node has been defined with interfaces, network objects created, and routers configured, the StoneGate management system can determine which way packets are flowing. Each interface in a firewall node has an ID associated with at least one IP address. If an interface receives a packet with a source address which is not a valid address for the networks connected to that interface, the packet will be dropped. The firewall has enough information about the directional flow of the packets and the networks to create rules eliminating the most improbable locations for any particular packet to be. The administrator can select **Anti-spoofing view**, in the Network Element Manager, to examine the anti-spoofing measures automatically created by the firewall. Illustration 4.3 shows an example anti-spoofing view in StoneGate’s GUI.

ILLUSTRATION 4.3 The StoneGate Anti-Spoofing View



Advanced StoneGate Routing

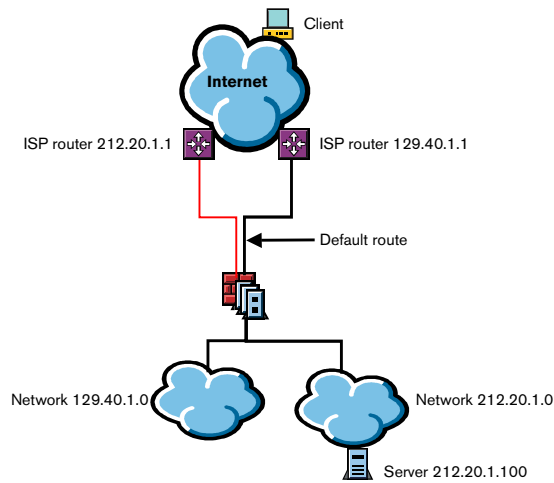
StoneGate features certain additional routing features that enable to configure the routing flexibly. StoneGate creates its routing tables automatically on the basis of the administrator input in the Routing View, so that even complicated routing solutions can be handled in a user-friendly way.

Policy Routing Entries

In addition to routing based on the destination IP address, the administrator can enforce the traffic with certain source IPs to be routed statically via a given gateway. This is called *policy routing*. Imagine a case like the one pictured in Figure 4.2. In the example there are two networks with routable, public addresses protected by StoneGate. Let's say these networks have public servers on them and no network address translation (NAT) is in use. These networks are served by a dedicated ISP each. However, the ISP router at 129.40.1.1 is defined as the default gateway. This means that in order to route

traffic from the 212.20.1.0 network correctly, we must define a static policy routing entry stating that traffic with 212.20.1.x as source IP will be routed via the ISP router 212.20.1.1.

FIGURE 4.2 *Policy routing example*



Policy routing can be configured easily in the GUI. In the Routing View, right-click and select **Policy Routing Entries**. Enter the appropriate source and destination IPs and netmasks and the correct gateway IP (see Illustration 4.4).

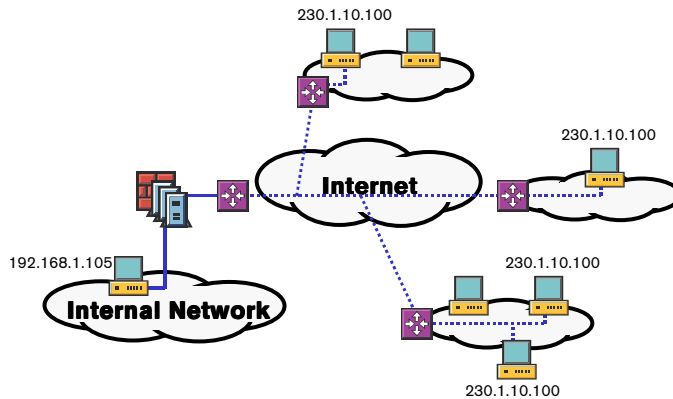
ILLUSTRATION 4.4 *Policy Routing entries*

The example above doesn't feature StoneGate's Multi-Link technology. Thus, static policy routing entries are required. If you were to use Multi-Link, the policy routing would be handled automatically on the basis of the NetLink elements. Multi-Link technology and NetLinks will be covered in the *StoneGate Advanced Implementation and Beyond* course.

Static IP Multicast Routing

IP multicasting is the transmission of an IP datagram to all hosts in a multicast host group, which is identified by a single destination IP address. This way the data needs only to be sent once to the multicast IP address, instead of sending it to all the hosts individually. A specific subset of Ethernet MAC addresses is reserved for mapping the IP multicast addresses (Layer 3) to multicast MAC addresses (Layer 2). For more information on multicasting, please see *Appendix B*, "Multicasting".

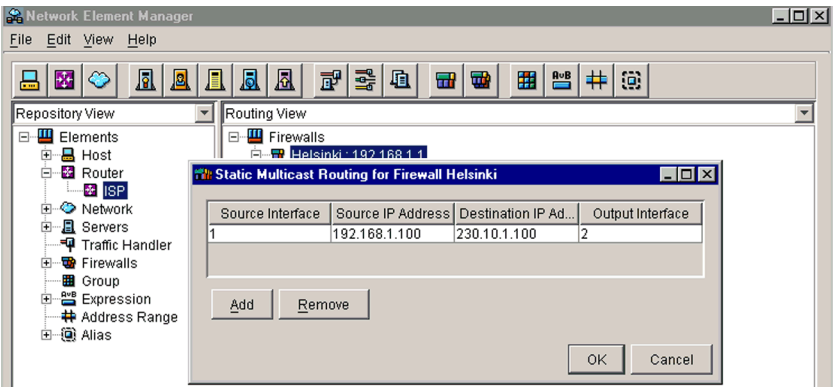
A simplified multicasting example is illustrated in Figure 4.3. The host 192.168.1.105 on the internal network sends multicast destined to 230.1.10.100. The routers forward the traffic through the Internet to the host group members on the different networks.

FIGURE 4.3 *IP multicast example*

StoneGate 2.0 implements *static* IP multicast routing, making it possible to relay multicast traffic through a firewall in a controlled way. As the multicast configuration is static—i.e., not relying on IGMP messaging—it is required to be explicitly defined by an administrator. This allows maintaining the control over the multicast traffic relayed through the firewall. Because of the static nature of the configuration, static IP multicast routing is suitable for enduring configurations, such as mutually agreed multicast traffic between organizations.

Static IP multicast routing is configured in StoneGate by defining static routes for multicast traffic. A StoneGate firewall listens to a defined input interface for multicast traffic coming in from a source IP address defined for an IP multicast route. Multicast traffic is handled in the firewall as any other traffic, according to the configured security policy. Finally, multicast traffic is sent through the defined output interfaces to the destination multicast IP address.

ILLUSTRATION 4.5 *Static IP Multicast Routing*



Summary

In this chapter, we examined two automated technologies employed by StoneGate to make a network administrator’s job easier. By generating routing tables based on defined network objects, and creating anti-spoofing measures through intelligent processing of network objects, the StoneGate firewall engines enable a more exact and structured security policy. Misconfigurations, complex routing tables, insecure routing protocols, and the potential to leave out critical anti-spoofing measures are mistakes of the past.

Review Questions

- What additional routes does StoneGate create automatically?
- Explain IP spoofing attacks, and why firewalls can be vulnerable.
- How does StoneGate defend your network against spoofing attempts?

Creating Basic Policies

What are quantum mechanics?
I don't know. People who repair quantums, I suppose.
-Terry Pratchett

StoneGate enables administrators to translate written corporate security policies into firewall security policies that control network traffic and access to network resources. Improving upon existing firewall technologies, StoneGate introduced several new concepts in security policy creation as part of its *Intelligent Rule Base* technology; these new features will be covered in detail in the *StoneGate Advanced Implementation and Beyond* course.

This unit will discuss creating basic StoneGate security policies. The first steps are understanding the network elements and defining the resources that you want to protect and control. Then you can create policies to control access to those elements, including user authentication, log options, time-sensitive availability, and more. These steps and basic navigation of the Security Policy Manager will be covered in this chapter. For more guidelines on creating security policies, see Appendix A, “*Guidelines for Building Network Security*”.

Objectives

Upon completion of this unit, you should be able to:

- identify the different network elements available
- list the six actions available for rules in a normal rule base
- explain the purpose of the Rule Tag

- list the different columns in access rules and NAT rules.

Network Elements

Network elements are the basic building blocks of any security policy. In order to control access to network resources, it is necessary to build policies with network elements. StoneGate provides you with many types of elements, including special elements that allow for unique and special combinations of other elements. In this section we'll provide a brief overview of each element and the properties assigned to it.



To work with network elements—creating, modifying or deleting - them—you can use the Network Element Manager from the StoneGate Control Panel. To open the application, click the Network Element Manager icon in the Launchpad. To create a new element of a given type you can simply click the element's icon on the toolbar or use the contextual menu available when right-clicking on an existing element in the GUI.



Host

The Host element represents any computer or other device on a network. It can be a user's PC, a Web server, or a Unix workstation, for example. The Host element can also be used to represent a single IP address on a multi-homed system, or to represent a server in a DMZ, such as an FTP server or e-mail server.



Router

The Router element defines special network devices that have more than one network interface, and can forward packets between network segments. Router elements are used to create routing tables for the firewalls, and are also a necessary component for defining NetLinks (see below).

Servers

There are several specific server types that can be defined in StoneGate. Each of these performs particular roles in assisting StoneGate firewalls with controlling and managing network traffic. For policies involving other servers, such as Web or FTP servers, use the Host element instead.

Authentication server



Authentication server elements are used to define external servers that can perform authentication services. StoneGate supports RADIUS and TACACS+ as authentication protocols, and can use any third-party authentication server that supports either of those standards.

LDAP server



The LDAP element is used for external LDAP directory servers. StoneGate has a built-in database for storing user information, which can be used for user authentication. However, if a corporation has an existing LDAP directory service, they can take advantage of the current infrastructure in the enterprise and use this element to specify their own LDAP directories.

Log server



Log Servers are a fundamental component of the StoneGate management system. They receive, process, store and retrieve log data from the firewall engines. During the installation of the management system, one or more Log Servers may have even been predefined; more Log Servers can be added to distribute the load.

Content Inspection Server (CIS)



Content Inspection Servers (CIS) define servers that perform virus-scanning, Web filtering, or other actions based on the content of the packets. This element defines the servers to which packets should be forwarded for content inspection and the protocols inspected.



DNS server

The DNS Server element is used to define DNS servers for inbound traffic management. These servers must support dynamic DNS updates, and are used in relation to the Server Pool element to enable load balancing of inbound traffic.

Traffic Handlers

Traffic handlers are a special set of elements used for inbound and outbound traffic management. With NetLinks, NetLink Pools and Server Pools you can take advantage of the advanced multi-link and load balancing features of StoneGate.



NetLink

The provider of some form of network connectivity, usually an ISP, is represented by the NetLink element. NetLinks are used like routers, but have additional properties, such as a list of IP addresses to probe in order to verify the operation of an ISP's connection.



NetLink Pool

A NetLink Pool is a collection of NetLinks, for use with outbound traffic management. NetLinks, NetLink Pools and their use with outbound traffic management are covered in the *StoneGate Advanced Implementation and Beyond* course.



Server Pool

Defines a group of servers, such as Web servers that are used for server load balancing or inbound traffic management. For Server Pools, information on the NetLinks to use, the servers that are part of the pool, and the DNS information for inbound traffic management are specified with this object.

Network



The Network element enables an administrator to define a network IP address range, by simply specifying the network address and subnet mask. Network elements make rule creation easier, as rules that apply to an entire network do not require the creation and use of an element for each device on that network. An *Any Network* element is available as a default element, and StoneGate automatically generates other network elements based on the firewalls' interface configuration.

Firewalls

Firewalls are a special class of element representing all of the StoneGate firewall engines administered by this StoneGate management system.

Single Firewall



A single firewall engine, which can be used to control access to network resources.

Firewall Cluster



A collection of two or more (up to 16) engines that load balance connections and provide high availability access control to network resources. The nodes of the cluster function as one virtual entity.

Group



An element that allows other network elements to be collected together into a single object to facilitate administration.

Expression



A special element that allows other elements to be combined with logical operators to create simple objects that represent complex sets of network resources.



Address Range

A set of IP addresses that can be used for some common purpose, but is perhaps smaller than a network segment.



Alias

A special element that can be used to represent a dynamic value. The value of which is determined upon where the alias is installed. Aliases are further explained in the *StoneGate Advanced Implementation and Beyond* course.



.....

Note: *You can add an informative comment explaining an element, its location or use in its Properties window, which is available by right-clicking the required element in the GUI.*

.....

You can also define categories and assign them to different elements as you see fit, for easier administration. Since StoneGate allows for centralized management of many different firewalls, you can group all of the elements for the Boston office under the `Boston` category, for example, and all of the elements for the Helsinki office under the `Helsinki` category. These categories can then be used in the Search View to more easily interpret the elements you see.

Rules

The sets of access rules and network address translation (NAT) rules contained in your rule bases constitute the operative element to implement your network security policy. In the rules, you determine the conditions for accepting or rejecting packets, translating IP addresses, and load balancing inbound and outbound traffic. Access rules take care of the basic traffic filtering while NAT rules help in concealing the internal network structure and support load balancing

of traffic. One of the crucial issues in designing the rule bases is the order of the rules; please see Appendix A, “*Guidelines for Building Network Security*” for further information.



You can define your security policies in the Security Policy Manager which is launched from the StoneGate Control Panel. Click its icon in the Launchpad. On the left panel of the Security Policy Manager, you will see the currently available rule bases. To edit any of these, right-click it and select **Open** from the contextual menu, or click the Open toolbar icon. To create a new rule base, click the New icon on the toolbar. In the opened New Rule Base dialog box, you can define the type and name of the new rule base. You need also to select the *template* on which the new one will be based. A default template is always available, but you can also define new templates. After creating the required security policy you can install it. You can install the same policy on several firewall systems under your administration at the same time.



Note: *Whenever you update a configuration of an element you need to reload your security policy on the firewall to make the changes effective. You can simply select the appropriate firewall system on your Control Panel and select **Refresh Policy** in the contextual menu.*

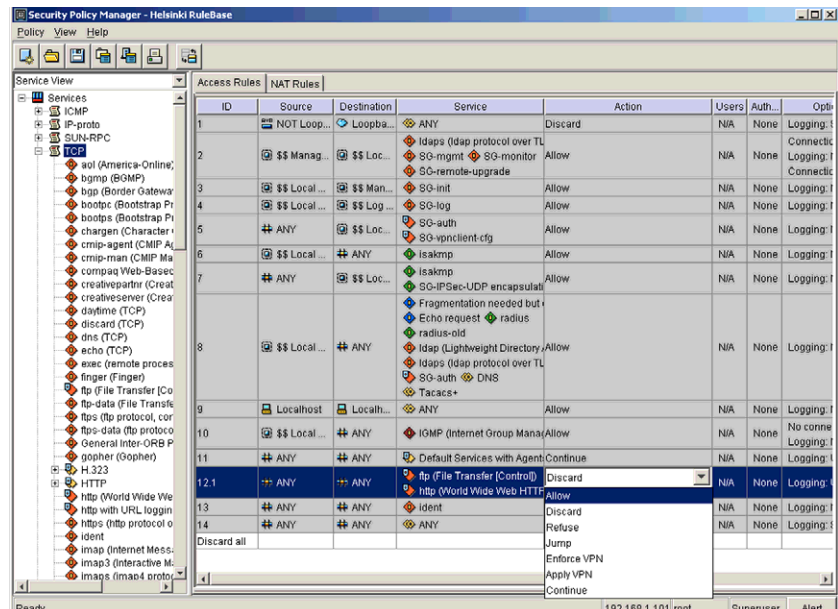
Different aspects of advanced rule base management are covered in detail in the *StoneGate Advanced Implementation and Beyond* course.

Access Rules

Access rules are defined in the Access Rules Editor of the Security Policy Manager. To create or modify rules, open the required security policy in the Security Policy Manager. You can define several

parameters for each rule. The use of different types of elements facilitates the design of complex rule bases.

ILLUSTRATION 5.1 *Example access rules*



ID

When added to a rule base, each rule is automatically numbered with a sequential ID that indicates the rule's position with respect to the existing rules. For instance, the ID determines whether the new rule will be processed first, 12th, or 500th in order in the rule base.

Source and destination

The source and destination specified by a rule are compared to the IP addresses in the packet headers under examination. Based on these and other criteria, the rule is applied to matching packets. The elements defined either as source or destination can be dragged to these cells from the left panel in the Access Rules Editor.

Service

Specific services can be defined for rules. The services can be dragged to this cell from the Service View on the left panel of the Access Rules Editor. The services available for rule design are categorized under ICMP, IP-Proto, SUN-RPC, TCP, UDP, and Group. Certain services are associated with Protocol Agents.

Action

The options for the action to be taken when a firewall detects a match for a packet are:

- **Allow:** the connection is allowed to pass through the firewall
- **Discard:** the connection is discarded
- **Refuse:** the connection is discarded and an ICMP error message is sent in response
- **Jump:** matching is continued in a specified sub-rule base until a match is found. If there's no matching rule in the sub-rule base, the process is resumed in the main rule base.
- **Enforce VPN:** the connection is allowed if the specified VPN is used; otherwise the connection is discarded
- **Apply VPN:** the connection is allowed if the specified VPN is used; if not, the rule is considered as non-matching and the matching process is continued
- **Continue:** all options specified in a rule with this action are stored in memory while the matching process continues. The options can, however, be overridden by a later, matching rule.

Users

Users and user groups can be specified in a rule to enforce user authentication for specific traffic. This adds granularity to the rule design. Defined users and groups can be dragged from the User View on the left panel of the Access Rules Editor.

Authentication

The authentication parameters and services to be used in a rule are specified in this cell. Available authentication services can be dragged from the Authentication Service View on the left panel. You can define the required authentication method and the authorization type in the Authentication Parameters window which is opened by double-clicking the Authentication cell.

Options

You can specify certain options for a rule in the Rule Options window which is opened by double-clicking the Options cell. For example, you can set connection tracking, time-out, and logging parameters for the rule.

Time

By double-clicking this cell, you can open the Rule Validity Time dialog box, in which you can set the time during which the rule will apply. The times you specify here are in Greenwich Mean Time (GMT), and will need to be adjusted to compensate for the firewall's local time zone.

Comment

You can write an informative comment describing the rule in this cell.

Rule Tag

The rule tag is the rule's permanent and unique identification, unlike the ID which changes each time a rule is added or deleted, the rule tag always remains the same. For instance, when viewing log entries in the Log Browser this helps to determine which rule has produced which entry.

NAT Rules

Network address translation rules form an important aspect of traffic management. NAT will be covered in Chapter 7 (*Basic Network Address Translation*) in this book.

ILLUSTRATION 5.2 *Example entries in NAT Rules Editor*

Access Rules		NAT Rules				
ID	Source	Destination	Service	NAT	Used on	Comm
3.1	Internal Server	ANY	ANY	Source: Static from Internal Server	ANY	
3.2	ANY	Internal Server	ANY	Destination: Static from Internal Set	ANY	
3.3	net-192.168.1.0/24	ANY	ANY	Source: Dynamic to 212.20.1.50/32	ANY	

You can design NAT rules in the NAT Rules Editor, which is accessed by opening a security policy in the Security Policy Manager and clicking the NAT rules tab. You can define several parameters for each NAT rule, as in the example in Illustration 5.2. The use of different types of elements facilitates the network traffic management in NAT rules. Note that NAT rules are applied only after a packet matches an access rule and is allowed by the firewall.

ID

When added to a rule base, each rule is automatically numbered with a sequential ID that indicates the rule's position with respect to the existing rules. For instance, the ID determines whether the new rule will be processed first, 12th, or 500th in order in the rule base.

Source and destination

The source and destination specified by a rule are compared to the IP addresses in the packet headers under examination. Based on these and other criteria, the rule is applied to matching packets. The required elements can be dragged to these cells from the left panel in the NAT Rules Editor.

Service

Specific services can be defined for rules. The services can be dragged to this cell from the Service View on the left panel of the Access Rules Editor. The services available for rule design are categorized under ICMP, IP-Proto, SUN-RPC, TCP, UDP, and Group. Certain services are associated with Protocol Agents.

NAT

The actual network address translation method is specified by double-clicking this cell. In the opened dialog box, you can define whether the rule will use source or destination translation. You can also set outbound load balancing parameters in this dialog box.

Used on

You can define here the firewalls to which the NAT rule in question shall apply. You can simply drag the required firewalls from the Repository View on the left panel to this cell.

Comment

You can write an informative comment describing the rule in this cell.

Summary

Firewall security policies enable system administrators to control specific network traffic and access to network resources. A security policy starts by defining the network elements, which correspond to the resources that you want to protect and control. The network elements and services function as building blocks for the rules that constitute your network security policy. The security policy is formalized in rule bases, which are collections of rules. They consist of access rules and NAT rules. The former take care of traffic filtering and the latter of translating IP addresses and supporting load balancing.

Review Questions

- List at least six network elements.
- What are the actions that the firewall can perform on a packet?
- What is the difference between Apply VPN and Enforce VPN actions?
- What is the purpose of the rule tag?

Basic Log Management

It is a mistake to think you can solve any major problems just with potatoes.
– Douglas Adams

This unit describes the tools and means of log management. It focuses on appreciating how different log views can be used to benefit administration and log management.

Objectives

Upon completing this unit, you should be able to:

- explain the fundamental concepts of log management
- list the five basic logging options
- describe and use log management tools and components
- formulate a log management strategy.

Basic Log Management Theory

Logs record the type and volume of inbound and outbound traffic. They provide a common point from which to access data about all network components. Accordingly, they are the fundamental resource for checking and proving your system. Although firewalls are capable of generating large amounts of log data, efficiently managed data can be used by administrators in many valuable ways.

For example, logs are useful when troubleshooting common network issues. StoneGate log filtering tools make it easy to recall data

highlighting the specific traffic or connections that may be causing trouble. In StoneGate, log data is used for effective network administration.

In addition, logs enable administrators to analyze network traffic such as FTP requests or visitors to the corporate intranet. This enables network administrators to justify the need for extra resources or other actions to make the service more functional. And, it helps administrators best use StoneGate security features such as intrusion detection.

When intruders target your system, log entries document their methods, the frequency with which they attack, and can alert you when automatic internal thresholds are tripped. Once you've found a malicious intruder, in order to press criminal charges you need evidence. Log entries can prove your case.

Fundamental Concepts

It is very important for StoneGate administrators to understand the functionality and use of logs in order to fully benefit from the extensive log data available. This section describes the terminology and concepts that are fundamental to StoneGate log management.

Log Data

Log data is the sequence of log entries. It includes all the documentation StoneGate records about inbound and outbound traffic. Log data can be filtered for different requirements and effective usage. The amount of data stored and available for viewing can also be configured by the administrator.

Log filtering and configuring tools can make or break the value you get from the log files in your system. The amount of log data can be enormous. It takes professional skill to manage the data so that only

relevant information is stored, and that it is available when needed and organized to be useful and easily understood.

Log Entry

Log Entries are records of packets that pass through the firewall. Each packet can be logged, but that takes up unnecessary memory and bandwidth. Most traffic does not pose a threat to network security.

In addition, log entries do not necessarily need to be kept forever. Entries older than certain time, standard HTTP requests to the Web server, and similar routine entries can be costly to archive. Therefore, it is often beneficial to delete routine or expired entries to avoid slowing the system or consuming too much server space.

Log Server

The Log Server processes log data and forwards it for storage to the database. If any client processes request the data, it is delivered immediately.

Filters

Filters allow administrators to speed up both data delivery and processing by limiting their requests to data that match a specific profile. To do so, administrators can create filtering profiles that specify the data to be retrieved, say, for browsing or exporting.

Logging Options

Before log data of traffic handled by the firewall can be viewed or used in any meaningful way, the logging settings of the firewall must first be configured. This is done by the firewall engine based on the logging criteria specified in the Security Policy Manager. Double-click on the Options cell of a rule to open the Rule Options dialog box.

By default, a rule's logging option is set to **Undefined**. In case there is a prior matching rule with the action set to **Continue** (see section *"Action"* in *Chapter 5* on page 89), a rule with the Undefined logging will use the options set in the Continue rule whenever it matches as well. In case there isn't a prior matching rule, the Undefined setting means that by default the log entry will be *stored* in the database. The other logging options listed below always override any options set in prior matching Continue rules. Hence, when defining the logging options you must first select the checkbox **Override collected values set with "Continue" rules** in the rule's Options cell before you can define one of the settings below.

Alert

The **Alert** setting generates an alert and a log entry every time the rule is triggered. This type of log entry is always stored. The alert notifications can be configured in the Alert Notification Manager.

Stored

The **Stored** setting generates a log entry and stores it to the database.

Essential

The **Essential** setting is used to control how logs entries are generated and stored when the firewall engine is running out of disk space.

When the firewall is *not* running out of disk space, the Essential setting acts just like the Stored setting. For example, assume that the firewall engine has lost its connection to the Log Server. Rather than lose log data, the firewall engine will store the log data itself while waiting for the connection to the Log Server to be restored. However, the engine cannot do this indefinitely. At some point it will begin to run out of disk space. This is where the Essential setting comes into play.

When the Essential setting is used, the firewall engine will save only those log entries that are marked as essential.

Transient

Log entries with the **Transient** setting are only available for immediate display and are not stored.

None

The **None** setting does not generate any log entry at all.

Log Accounting

You can also enable the generation of log accounting information for connections. This log data is generated at the end of a connection, and it contains information on the elapsed time, and the number of bytes sent and received. Log accounting can be enabled by selecting the **Log Accounting Information** radio button in the Rule Options dialog box.

Log Management in Practice

Once log data is generated by the firewall engines, the StoneGate administrator has many options regarding what happens to the data. The data can be viewed, stored, discarded, filtered, pruned, exported, archived, or many combinations of the above. StoneGate provides the administrator with a number of tools to make this data management simple and efficient.

Log Browser



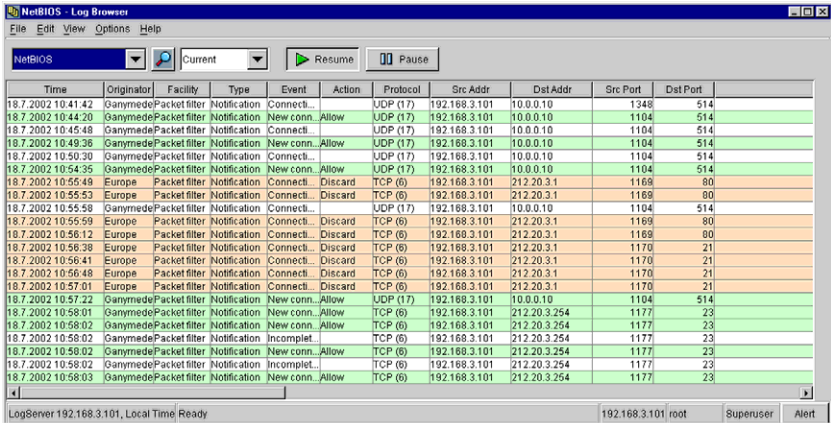
The Log Browser allows administrators to view log data generated by the firewall engines. The administrator may either view fresh entries as they arrive from the server or browse stored, historical data. Log data can be retrieved from the database sorted by timestamp, with entries starting either from the first or the last second of interest. In addition,

it is possible to view also archived files in the Log Browser. In summary, there are three different options, or modes, for viewing the logs:

- **Current:** showing always the most recent log entries including the ones generated by rules with the logging option *Transient*. The display of log data on the screen can be controlled with the **Pause** and **Resume** buttons.
- **Database:** showing stored entries retrieved from the log database. The data retrieval can be controlled with the start and end time settings and filtering profiles. Transient entries are not included.
- **Direct Archive:** showing entries in the archive files. The data retrieval can be controlled with the start and end time settings and filtering profiles.

Illustration 6.1 shows an example of the Log Browser display. The different columns can be shuffled to a different order or hidden from view according to your needs.

ILLUSTRATION 6.1 Log browser in Current mode



Time	Originator	Facility	Type	Event	Action	Protocol	Src Addr	Dst Addr	Src Port	Dst Port
18.7.2002 10:41:42	GanymedePacket filter	Notification	Connect...			UDP (17)	192.168.3.101	10.0.0.10	1348	514
18.7.2002 10:44:20	GanymedePacket filter	Notification	New conn...	Allow		UDP (17)	192.168.3.101	10.0.0.10	1104	514
18.7.2002 10:45:48	GanymedePacket filter	Notification	Connect...			UDP (17)	192.168.3.101	10.0.0.10	1104	514
18.7.2002 10:49:36	GanymedePacket filter	Notification	New conn...	Allow		UDP (17)	192.168.3.101	10.0.0.10	1104	514
18.7.2002 10:50:30	GanymedePacket filter	Notification	Connect...			UDP (17)	192.168.3.101	10.0.0.10	1104	514
18.7.2002 10:54:35	GanymedePacket filter	Notification	New conn...	Allow		UDP (17)	192.168.3.101	10.0.0.10	1104	514
18.7.2002 10:55:48	Europe	Packet filter	Notification	Connect...	Discard	TCP (6)	192.168.3.101	212.20.3.1	1168	80
18.7.2002 10:55:53	Europe	Packet filter	Notification	Connect...	Discard	TCP (6)	192.168.3.101	212.20.3.1	1168	80
18.7.2002 10:55:58	GanymedePacket filter	Notification	Connect...			UDP (17)	192.168.3.101	10.0.0.10	1104	514
18.7.2002 10:55:59	Europe	Packet filter	Notification	Connect...	Discard	TCP (6)	192.168.3.101	212.20.3.1	1168	80
18.7.2002 10:56:12	Europe	Packet filter	Notification	Connect...	Discard	TCP (6)	192.168.3.101	212.20.3.1	1168	80
18.7.2002 10:56:38	Europe	Packet filter	Notification	Connect...	Discard	TCP (6)	192.168.3.101	212.20.3.1	1170	21
18.7.2002 10:56:41	Europe	Packet filter	Notification	Connect...	Discard	TCP (6)	192.168.3.101	212.20.3.1	1170	21
18.7.2002 10:56:48	Europe	Packet filter	Notification	Connect...	Discard	TCP (6)	192.168.3.101	212.20.3.1	1170	21
18.7.2002 10:57:01	Europe	Packet filter	Notification	Connect...	Discard	TCP (6)	192.168.3.101	212.20.3.1	1170	21
18.7.2002 10:57:22	GanymedePacket filter	Notification	New conn...	Allow		UDP (17)	192.168.3.101	10.0.0.10	1104	514
18.7.2002 10:58:01	GanymedePacket filter	Notification	New conn...	Allow		TCP (6)	192.168.3.101	212.20.3.254	1177	23
18.7.2002 10:58:02	GanymedePacket filter	Notification	New conn...	Allow		TCP (6)	192.168.3.101	212.20.3.254	1177	23
18.7.2002 10:58:02	GanymedePacket filter	Notification	Incomplet...			TCP (6)	192.168.3.101	212.20.3.254	1177	23
18.7.2002 10:58:02	GanymedePacket filter	Notification	New conn...	Allow		TCP (6)	192.168.3.101	212.20.3.254	1177	23
18.7.2002 10:58:02	GanymedePacket filter	Notification	Incomplet...			TCP (6)	192.168.3.101	212.20.3.254	1177	23
18.7.2002 10:58:03	GanymedePacket filter	Notification	New conn...	Allow		TCP (6)	192.168.3.101	212.20.3.254	1177	23

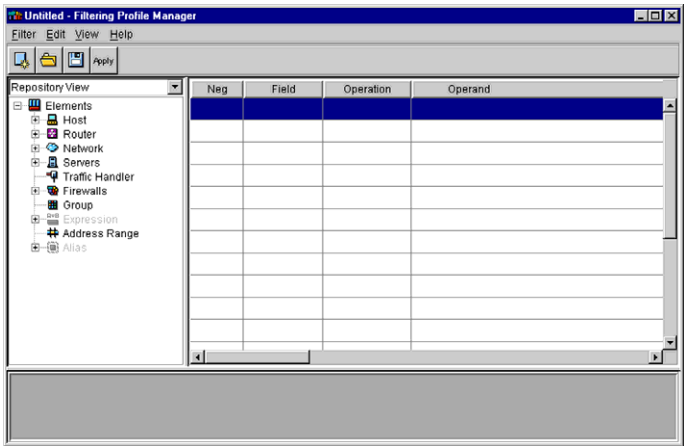
The Log Browser also allows administrators to view log data that fits specific criteria through the use of filtering profiles. By using these profiles, administrators can narrow their log searches, thereby saving time and resources. Custom filtering profiles can be created with the Filtering Profile Manager (see *below*).

Filtering Profile Manager



In the Filtering Profile Manager, Administrators can create filter profiles that specify precisely the type of log data that they want to filter out. Once created, these profiles can be named and saved and used for different log management tasks whenever they are needed. Filters can be used, for example, to display only certain type of network traffic in the Log Browser or to export certain kind of log entries while deleting others.

ILLUSTRATION 6.2 *Filtering Profile Manager*



Log Pruning Filter Manager

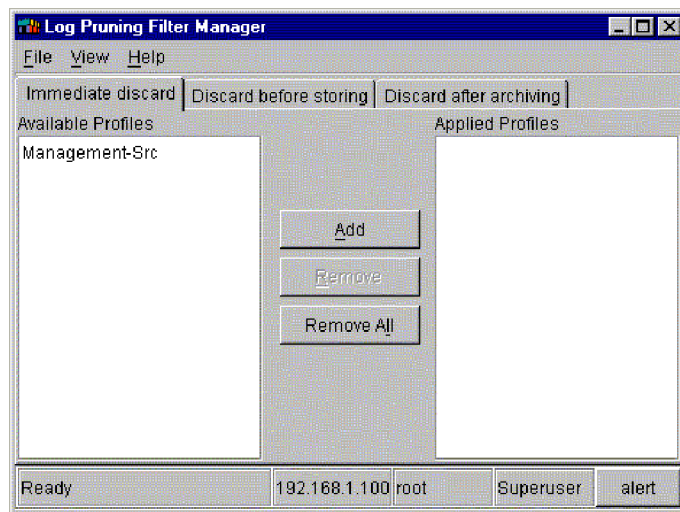


The Log Pruning Filter Manager, allows administrators to further manage log data by defining when data can be discarded. Three options are available.

The first option, **Immediate Discard**, allows the administrator to discard data before it is sent to the Log Browser. The second option, **Discard Before Storing**, allows the administrator to view the data in the Log Browser, but discard it before storing it in the database. Finally, **Discard After Archiving** allows the administrator to forward data directly to the archives.

Administrators can configure the Log Pruning Filter Manager by identifying specific Filter Profiles inside the Log Pruning Filter Manager and whether the log data generated by those filters should be discarded immediately before storing or after archiving. However, access to the Pruning Filter Manager is reserved to the Superuser. This is done to protect sensitive log data from misuse.

ILLUSTRATION 6.3 *Log Pruning Filter Manager*



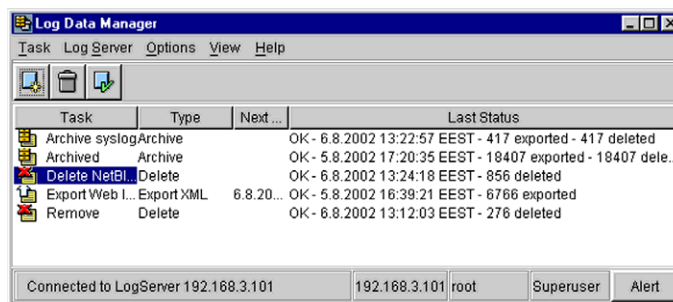
Log Data Manager



The Log Data Manager allows the administrator to further manage log data by deleting old data that is no longer required or by archiving log data that is not currently relevant. To do so, the administrator

creates and configures data management tasks that can be run either manually or automatically. In this way the administrator can ensure that the amount of log data does not exceed the system's capacity or unnecessarily utilize its resources.

ILLUSTRATION 6.4 *Log Data Manager*



Summary

Logs perform many functions. They simplify and facilitate administrative tasks that detail system conditions from several perspectives. Generally speaking, logs are used to:

- record the type and volume of inbound and outbound traffic
- troubleshoot network issues
- provide evidence for necessary network upgrades
- provide evidence of network attacks
- track actions and connections such as HTTP and FTP.

This chapter reviewed the following components of the StoneGate log administration system:

- Log Browser
- Filtering Profile Manager
- Log Pruning Filter Manager
- Log Data Manager.

Review Questions

- List three different uses for log data.
- Where does the administrator define log generation options?
- What are the five options available for log generation?
- What is the main purpose of filtering profiles?

Administrator Management

The most ineffective workers are systematically moved to the place where they can do the least damage—management.

– Scott Adams, *The Dilbert Principle*

This unit describes the process and means of administrator management. StoneGate enables you to designate administrative tasks and responsibilities for the best use of available resources.

StoneGate networks are checked at three levels to minimize the potential for human error and promote administrative flexibility:

- the system checks the administrator type and associated rights
- the Management Server checks the grant list of elements that fall under the administrator's rights
- the Management Server checks for cross-referencing relationships between the elements defined by a security policy.

Objectives

Upon completing this unit, you should be able to:

- define the three administrative levels
- list the rights that apply to each level
- plan and implement a system that incorporates various administrators in a StoneGate environment with multiple sites and StoneGate engines.

Administrator Accounts

Administrator accounts are divided into three levels: Superuser, Editor and Operator. Each level is entitled to certain rights and permissions within the StoneGate environment. Some of these permissions are specified by the user, while others are set by default. In any case, the network administrator will have to decide how to delegate and assign various responsibilities among the administrative staff. This is achieved with the various administrator accounts which are configured in the Administrator Manager.

Issues to Consider

When deciding how to delegate responsibility in a network, a network administrator needs to carefully determine the type of account to assign to each administrator. To do so, the network administrator must consider the following issues:

- What does the administrator *need* access to?
- What doesn't the administrator *need* access to?
- What skills, understanding, and experience does the administrator have? For example, can this administrator safely cope with tasks the system enables him or her to access, or is there a risk this person may misuse or misunderstand those rights?
- Does any network segment reference an area the administrator is responsible for? Remember to consider whether cross-reference relationships will circumvent the rights that should limit an administrator. (For more detailed information on cross-referencing, see "*Cross-reference checking*" on page 113.)
- Is there any part of the StoneGate environment that should be protected from the particular administrator to prevent harm to the system?



.....

Caution: Administrator passwords must be carefully selected. They should be long enough (eight characters minimum) and contain combinations of alphabetical, numerical and special characters. Avoid using any part of your or your relatives' names, birthday, home address or similar—not even spelled backwards.

.....

Superuser, Editor and Operator

The three types of administrator accounts broadly determine individual assignments and restrictions. See Table 7.1 on page 113 to view a summary of administrator rights, restrictions and conditions regarding element manipulation. Table 7.2 on page 115 sums up the permissions for maintenance operations.



.....

Tip: To ensure proper auditing of security policy creation and installation, you should create unique accounts for every administrator of the system, rather than having “generic” accounts with multiple users.

.....

Superuser

Superuser is the highest StoneGate administrator class, and confers almost complete system access. Superusers can create or update any element without restriction. They can also create and update other administrator accounts, and are the only administrators that can do so. By default, one Superuser is created during StoneGate installation, although more can be added later.

Superusers also determine network architecture. They define and plan the StoneGate environment, sites, VPNs etc. They are responsible for

the entire security policy in firewall clusters and for the operational functionality of the system. Superuser is the only account that has access to the Audit Manager which used for tracking actions performed and events occurred on the system.



.....

Caution: Because StoneGate bypasses permission checks on the Superuser account, and because the Superuser controls all other Superuser accounts, the number of Superusers should be kept low.

.....

Typically, a StoneGate network should have a single Superuser. Although it may be convenient to have a “spare” Superuser in the event the normal Superuser administrator is absent for a long period of time (e.g., vacation), most modifications requiring a Superuser can typically wait until the usual Superuser is available. If it is necessary to have more than one Superuser, the number should be kept as low as possible.

Editor

Editors have restricted rights and perform well-defined administrative tasks. For example, Editors may be assigned the administration of a particular site or a network segment.

Editors can create and update all simple elements, provided a cross-reference check permits it. They can create and update granted elements if they are on the grant list of the element, and if a cross-reference check permits it. They can also perform installation operations on firewall clusters if their ID is registered on the grant list. (For more detailed information on simple and granted elements, see page 109.)

Editors are typically the local persons in charge. Because they have limited rights to the network, these accounts are best used for local

management or for distinct parts of the network. However, when adding Editors to grant lists and when assigning administrator accounts, special consideration should be given to matching abilities and responsibilities with access levels and permissions. Too much access can be as difficult to manage as too little access—and it's often more difficult to restrict access after damage is done than to increase privileges as experience warrants.

Operator

Operators have the fewest rights of all administrator accounts. They have *read only* access to all simple elements, and have access to granted elements only if registered on the grant list. They can also perform installation operations on firewall clusters if registered on the appropriate grant list. Operators are mainly system spectators. They can check that configurations are correct and forward information to the appropriate parties.

Elements

Elements define the way the network identifies any hardware or software that can be addressed. Elements themselves are divided into two categories: simple elements and granted elements. Each category is defined by the rights an administrator needs to create and manage them.

Simple Elements

Simple elements include any hardware or software that can be directly addressed, and is recognized by the system. Specifically, they do not have any cross-reference relationships that could circumvent the rights that should limit an administrator.

Simple elements include the following:

- simple network elements

- combined network elements
- special network elements *except* firewalls (i.e., routers and servers)
- users and groups
- services.

Managing simple elements

Simple elements have simple relationships. They can be created by administrators who have the appropriate rights. Nothing more is required to manage or alter a simple element, although deleting a simple element is impossible as long as another element is referencing it.

A host, for instance, may be a simple element. However, the host might be referenced in several rule bases. Let's assume that the administrator that created the host later decides to delete the host. Before StoneGate allows the administrator to make this change, the Management Server will check the administrator's privileges for that particular host. In addition, however, the Management Server will also check whether the administrator has permission to update each element that references the host. If not, then the administrator is denied the authority to delete the host element. For further explanation of this concept, see "*Permission Checking*" on page 111.

Granted Elements

Granted elements include:

- firewall clusters
- single firewalls
- template rule bases
- normal rule bases

- sub-rule bases.

When they are created, granted elements are assigned to a restricted list of administrators. Later, Superusers or Editors on the list can add administrators to the list. By default, the creator of an element is also included in the element grant list.

Updating and Deleting

Updating the list of a granted element is equivalent to updating the element itself. Administrator rights alone are not sufficient for updating granted elements. Deleting a granted element is impossible as long as another element is referencing it.

Permission Checking

Permissions are not limited by administrator rights. This is because elements (each with its own permissions) may have multiple logical occurrences, some of which fall within, and some outside of, a certain administrator's rights.

Instead, permissions are limited and the administrator's scope is defined by grant lists and cross-reference checks. That means the actions of a StoneGate administrator are checked twice: by grant lists and by cross-reference checks.

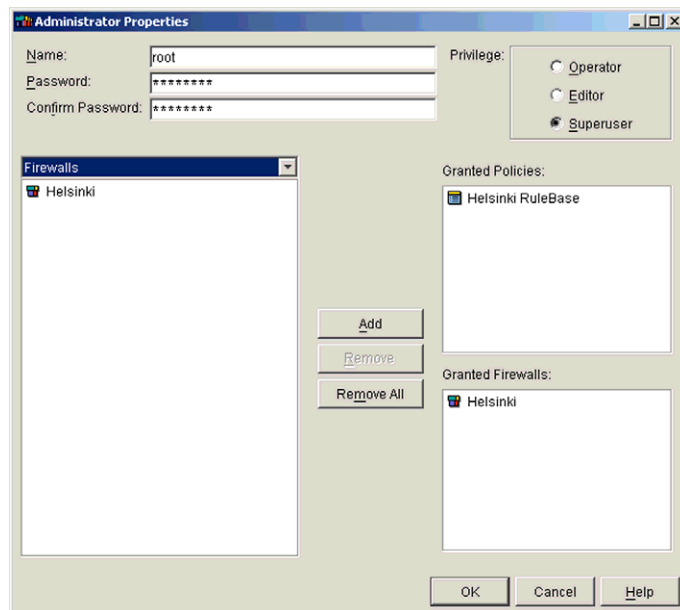
StoneGate permission checking is especially effective for multiple sites and firewall clusters. In complex network architectures like these, administrative responsibilities are distributed among local administrators, and no local administrator has enterprise-wide rights. StoneGate grant lists and cross-reference checks distribute administrative tasks and minimize the risks of serious and wide-ranging mistakes.

Grant lists

The *grant list* of a granted element specifies the administrators allowed to edit the element, taking into account the rights of that administrator. The checking process states that to update or delete the granted element, the administrator must be on the grant list of all granted elements that reference the element to be updated or deleted.

However, acknowledgement in the grant list does not affect or exceed the rights of any individual administrator. For instance, an Operator who is included on the grant list of a firewall cluster would only be allowed to read and install a security policy. Inclusion on the grant list does not imply the Operator can exceed the basic rights of an Operator. Grant lists should be designed in terms of distributing administrative responsibility.

ILLUSTRATION 7.1 *Example of a grant list*



Cross-reference checking

The *cross-reference check* tracks relations between elements. It notifies the administrator of the implications any proposed update will have on the overall security policy. This checking process states that to update (or delete) any element, an administrator must already have rights to make the same update on all elements referencing the element to be updated (or deleted).

For instance, consider an administrator who wishes to delete a server in a rule base. If the administrator has rights to update the server, but the grant list of this rule base does not mention the particular administrator, then the action is not allowed. This protects the system from changes being made by an unauthorized administrator. Cross-reference checking is very useful in complex networks that contain various StoneGate sites to be administered.

Permissions on administrative operations

The administrator rights on accessing and modifying elements are summarized below.

TABLE 7.1 *Administrator permissions for manipulating elements*

	Actions	Type of administrator		
		Superuser	Editor	Operator
Simple elements	Reading	Yes	Yes	Yes
	Creating	Yes, with name check	Yes, with name check	No
	Updating	Yes	Yes, if same permission on referencing elements	No
	Deleting	Yes, if not referenced	Yes, if not referenced	No

TABLE 7.1 *Administrator permissions for manipulating elements (Continued)*

	Actions	Type of administrator		
		Superuser	Editor	Operator
Granted elements	Reading	Yes	Yes	Yes, if on grant list
	Creating	Yes, with name check	Yes, with name check	No
	Updating	Yes	Yes, if on grant list and same permission on referencing elements	No
	Deleting	Yes, if not referenced	Yes, if on grant list and not referenced	No

Permissions on maintenance operations

Several maintenance operations are available to administrators with some restrictions depending on their privileges.

- **Creating administrator accounts:** Only Superusers are allowed to create and manage administrators accounts. They can also upgrade or downgrade the privileges of registered administrators.
- **Adding an administrator to the grant list of a granted element:** Only Superusers can add another administrator to grant lists without restriction. Editors are allowed to the same operation if they are included in the grant list of the element, but Operators cannot.
- **Viewing logs:** All administrators are allowed to use the Log Browser to visualize logs using any of the available filters.
- **Viewing audit data:** Only Superusers can manage audit data recorded on the system events and administrator actions.
- **Applying pruning filter:** Only Superusers are allowed to apply pruning filters with the help of the Log Pruning Filter Manager.

- **Updating Routing View:** In the Routing View, any instance of network elements can be added or removed by Superusers and Editors provided that they are on the grant list of the firewall. Operators have no permission to do so.
- **Installation of a security policy on the cluster:** All administrators can install a rule base on the firewall cluster provided they belong to the grant list of the firewall. The rule bases available for installation on a given cluster can be specified by Superusers and Editors.
- **Monitoring commands:** All administrators are allowed to set the firewall online or offline. Editors and Operators must be on the grant list of the firewall cluster to do so.

The following table summarizes all maintenance operations that are allowed to the different types of administrators.

TABLE 7.2 *Maintenance operations allowed to administrators*

Maintenance operations	Administrator type		
	<i>Superuser</i>	<i>Editor</i>	<i>Operator</i>
Administrator accounts	Yes	No	No
Adding administrator to grant list of rule base	Yes	Yes, if on grant list	No
Viewing logs	Yes	Yes	Yes
Viewing audit data	Yes	No	No
Applying Log Pruning Filter	Yes	No	No
Updating Routing View	Yes	Yes, if on grant list of firewall	No
Rule base installation on a firewall / Policy Refresh from Control Panel	Yes, if rule base installation granted on firewall element	Yes, if on grant list of firewall, and rule base installation granted on firewall element	Yes, if on grant list of firewall, and rule base installation granted on firewall element
Monitoring commands for firewall state	Yes	Yes, if on grant list	Yes, if on grant list



.....

Note: Two administrators cannot update the same element at the same time. The only exception is the rule bases, which are subject to a locking procedure when they have been opened.

.....

Summary

Rights and permissions are carefully managed in an effective network using several strategies. Administrator accounts are divided into three levels: Superuser, Editor and Operator. In turn, administrator rights on a StoneGate network are also checked at three levels:

- the system checks the administrator type and associated rights
- the Management Server checks the grant list of elements that fall under the administrator's rights
- the Management Server checks for cross-referencing relationships between the elements defined by a security policy.

Review Questions

- What are the three administrative levels?
- How are an administrator's permissions to modify or delete an element checked?

Network Address Translation (NAT)

The Tao can't be perceived. Smaller than an electron, it contains uncountable galaxies.

– Lao-tzu, *Tao Te Ching*

Network address translation (NAT) is arguably one of the largest roles a firewall plays in addition to enforcing corporate security policies. With the explosion in Internet usage in recent years, the 32-bit addresses used for TCP/IP were in danger of becoming exhausted. NAT was developed to alleviate the growing shortage of addresses. It enables network administrators to each use the same set of non-routable, or private, IP addresses, hiding entire networks behind a single routable, public IP address instead. In this unit, you will learn how StoneGate handles network address translation.

Objectives

Upon completing this unit, you should be able to:

- list the three main types of network address translation in StoneGate
- describe the difference between static and dynamic NAT
- provide at least one example of dynamic source translation
- provide at least one example of destination translation
- explain the need for proxy ARP (address resolution protocol) with NAT.

Network Address Translation Overview

Network address translation changes the source or destination IP address or port for packets traversing the firewall. It is often used to hide one or more internal networks behind a single, routable IP address on the external network. But address translation can also be used to translate an external, routable destination address back into the real, private internal address of a server. For destination NAT, it is also possible to perform port translation (sometimes referred to as PAT). Port translation can be used to redirect a standard service, such as HTTP (port 80), to a non-standard port (e.g., port 8080).

For network address translation in StoneGate there are six methods possible:

- static source translation (often a single IP address to another single IP address) [one-to-one relationship]
- dynamic source translation (translating one or more networks to a single IP address or a pool of IP addresses) [many-to-one/many-to-some relationship]
- static destination translation (often a single IP address to another single IP address) [one-to-one relationship]
- destination port translation (translating the port for the service/protocol to a non-standard one) [one-to-one relationship]
- combination of static source and destination translation
- dynamic source translation for outbound load balancing.

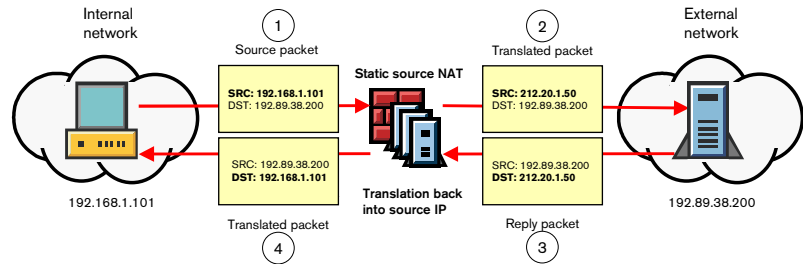
To get a better understanding of NAT in general, we'll look at examples for several of these methods.

Static Source Translation

In static source translation, the source IP address of a packet is changed to an IP address for another network. Often, the original source address is the actual assigned IP address for a device on an

internal network or DMZ, and is translated to a public IP address for an ISP's assigned network range for that company.

FIGURE 8.1 *Static source translation*



In Figure 8.1, the packet starts out with the source (SRC) and destination (DST) addresses as seen in step 1. The firewall replaces the source address of the packets with a new source address, as seen in step 2. Because StoneGate's Multi-Layer Inspection records information to track connections, it knows automatically to translate any reply packets received from the destination machine as well. So as the server in this example responds, the destination address in step 3 is replaced with the original address again in step 4, ensuring that the responses get back to the proper host.

Static source translation can be said to be a one-to-one translation. That is, each IP address is matched with a corresponding translated IP address. Therefore, you can set up a translation, where you specify a host's original address should be translated to an IP address, or to another host's IP address. You can also use static source translation to translate one network to another network, where every machine in the first is translated to the second. To translate networks, however, the netmask of the two networks must be identical, ensuring the proper number of IP addresses for the translation process.



Tip: If you attempt to set up an invalid translation, for example, a static source translation from a host to a network, StoneGate will provide a warning, and give you the opportunity to correct the problem.

Dynamic Source Translation

Dynamic source translation is very similar to static source translation, except the IP addresses of one or more networks or address ranges are translated to a single IP address on the other side. Dynamic source translation, sometimes referred to traditionally as *hide NAT*, is often used to mask the internal networks of a company behind a single public, routable IP address (provided by an ISP, for example).

FIGURE 8.2 *Dynamic source translation*

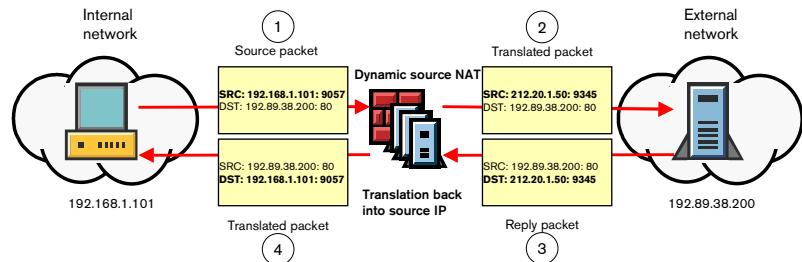


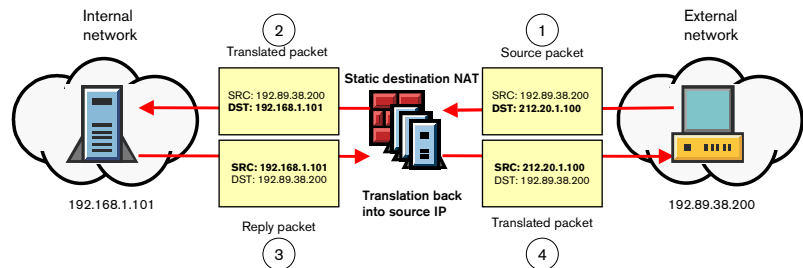
Figure 8.2 illustrates the process for dynamic source translation. Since dynamic source translation involves a many-to-one or many-to-some relationship (multiple hosts on the inside using a single address or a small pool of addresses on the other side), the firewall needs some additional information to differentiate the connections. To do so, the firewall also uses the source port, translating it as necessary, using the unreserved high ports to track each individual connection, as seen in steps 1 and 2. By doing so, it can use this information in the

connection tracking tables to automatically translate the reply packets back to the original source address and port, as seen in steps 3 and 4.

Destination Translation

There may be times when it's useful to translate the destination address instead of, or in addition to, the source address. StoneGate allows you to also perform static destination translation, for both IP addresses and ports. Destination translation is used for translating the addresses of Web and FTP servers, for example, from the Internet-accessible external address, to the real internal address on a DMZ.

FIGURE 8.3 *Destination translation*



In the example illustrated in Figure 8.3, a host on the Internet connects to a server on the internal network. The host connects to the external, public IP address in step 1. StoneGate then translates the destination address to the private IP address of the server on the internal network. The server sends its response back in step 3, where StoneGate automatically translates the source address back to the external IP address, shown in step 4.

Destination port translation

Destination translation can also be used to translate ports. For example, Web traffic to the corporate Web servers on a DMZ would typically come in on port 80. However, an administrator may wish to run the Web service on a non-standard port, such as 8080, for security

or administrative reasons. With StoneGate, the administrator can have the original destination HTTP port 80 translated to port 8080, using static destination port translation with or without destination address translation.



.....

Caution: The order of NAT rules should proceed from the most specific to the least specific. A host on an internal network with static source translation should have its rule before a dynamic source translation rule for all of the other hosts in that same network, otherwise the wrong translation may take place.

.....

The next section will cover some more examples of each translation method, and how they might be used in a hypothetical corporate environment.

Network Address Translation Examples

To help illustrate the use of network address translation in StoneGate, it is useful to explore some examples. Here you can read how an administrator can set up StoneGate to perform the various translations typically used in a corporate environment.

In a typical corporate network you would have at least one rule to perform static source translation of the mail server for SMTP traffic, so that it can properly send outgoing mail, while dynamic source translation is employed to mask the internal networks behind a single public IP address. Web or FTP servers and the mail server would also have destination translation, so that hosts on the Internet could reach them through other public IP addresses.



To set up network address translation, you use the NAT Rules Editor in the Security Policy Manager. The NAT Rules editor works much

like the Access Rules editor, with source, destination and service columns.

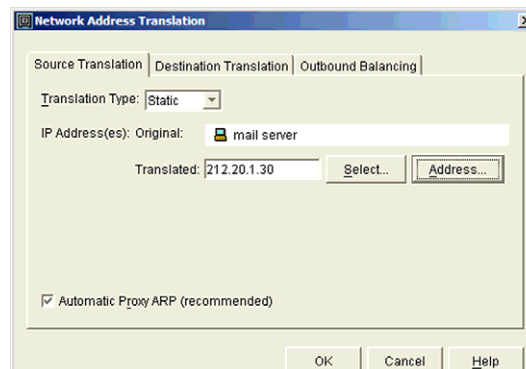


Note: For each NAT rule specified, there should be at least one corresponding access rule to allow the connection. Since NAT is applied only after the action from the access rule is performed, NAT rules without a matching access rule (for source, destination and service) are meaningless.

Static Source Example

To translate the source address for the mail server in this example, you create a NAT rule, where the source address is the host element for the mail server (using the private IP address). The destination address would be an element representing the Internet, typically an expression element of **Not internal** (the negation of all internal networks). For the service, specify the SMTP service.

ILLUSTRATION 8.1 Static source translation of an SMTP server



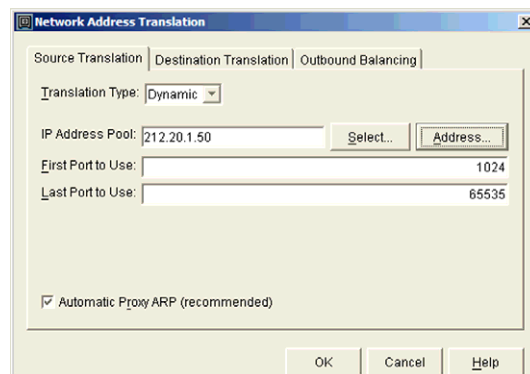
For the NAT column then, the Source tab can be used to set the static translation, as shown in Illustration 8.1. From the selection box, select

Static; StoneGate will automatically populate the original address as the mail server element, since that element was in the source column. For the translated address, you can then either use the **Select** tool to pick a defined network element, or use the **Address** button to enter an IP address.

Dynamic Source Example

Once the more specific NAT rules are in place, you can add the less specific ones, such as dynamic source translation rules. In this example, you can add a rule to dynamically translate the source address for elements in the internal network to a single external IP address or a small pool of addresses.

ILLUSTRATION 8.2 *Dynamic source translation of the internal network*



In Illustration 8.2, the translation type is set to Dynamic. Again, you can select a defined network element for the translation address using the **Select** tool, or use the **Address** button to specify a single IP address, or an IP address pool (range of addresses). The port range, used for tracking individual connections, is also customizable, but is set to the default non-reserved ports of 1024–65535.

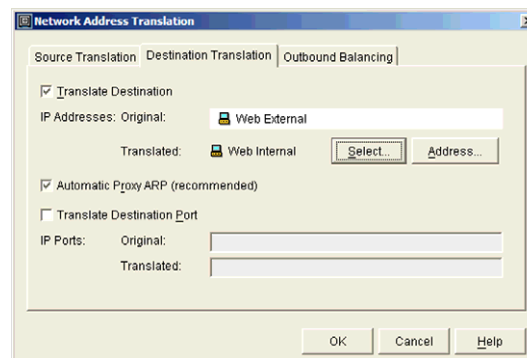


Note: Whenever possible, you should consider using a pool of IP addresses (two or more) for dynamic translation, since a larger pool will increase the performance of the firewall.

Destination Translation Example

To enable Internet traffic to reach the corporate Web server, static destination translation would be deployed. In this example, you would translate the external, or public, IP address of the Web server to the internal, private address. This translation method is commonly used for translating incoming traffic for services like HTTP, FTP, or SMTP.

ILLUSTRATION 8.3 Destination translation of a Web server



For this example, Illustration 8.3 shows the dialog box for destination translation. In destination translation you check the box **Translate Destination**. The destination address (in this case, the host element Web External, representing the public IP of the Web server) is then automatically placed in the **Original** box. You can then use the **Select** tool or **Address** button to pick the defined network element or enter the IP address to which the destination should be translated.



Tip: It is not necessary to set up pairs of translation rules for NAT, as StoneGate's Multi-Layer Inspection automatically performs the reverse translation for reply packets. However, you can translate both the source and destination addresses in a single rule if you require.

When you have the translation rules in the example set up, you have a NAT rule base as shown in Illustration 8.4.

ILLUSTRATION 8.4 Final NAT rule base example

Access Rules: NAT Rules						
ID	Source	Destination	Service	NAT	Used on	Comment
3.1	mail server	Not internal	smtp	Source: Static from mail server to 212.20.1.30/32	ANY	
3.2	internal	Not internal	ANY	Source: Dynamic to 212.20.1.50/32 1024-85535	ANY	
3.3	Not internal	Web External	http	Destination: Static from Web External to Web Internal	ANY	
No NAT						

Outbound Load Balancing NAT

The third option for translation, in addition to source and destination translations, is outbound load balancing. For StoneGate's Multi-Link Technology, the firewall translates outgoing connections to an address from a pool assigned to each available NetLink. These are used to measure response times and set each connection to use a particular network provider.

ILLUSTRATION 8.5 Outbound NAT rule example

Access Rules: NAT Rules						
ID	Source	Destination	Service	NAT	Used on	Comment
3.1	Finland Internal Network	Not internal	ftp (File Tran	Load balancing: NetLink Pool	ANY	
No NAT						

Illustration 8.5 shows an example Outbound NAT rule. Outbound traffic management using NAT with Multi-Link Technology is

covered in detail in the *StoneGate Advanced Implementation and Beyond* course.

Proxy ARP (Address Resolution Protocol) and NAT

Proxy ARP is a specification that allows a device to respond to ARP requests on behalf of some other device on the network. When network address translation is used on a firewall, the firewall is typically configured to use proxy ARP so that it can accept packets for the translation addresses. The firewall uses proxy ARP instead of requiring the administrator to assign all of the translation addresses to the firewall's network interface.

Traditional firewalls required the entry of all ARP entries for the translation addresses in the operating system, however. For large enterprises with many translation rules, this manual proxy ARP entry could lead to errors, and was time consuming to maintain and troubleshoot.

With StoneGate, proxy ARP is handled automatically. A check box for each translation type is checked by default to enable proxy ARP, although you have the option to uncheck it if you desire.



.....

Note: For some protocols, like FTP, source and destination address information is included in the data payload of the packets. Traditional firewalls either cannot support these protocols or have extreme difficulty in doing so. StoneGate's Protocol Agents can solve the problem because they can examine the data payload and modify it. Protocol Agents are discussed in more detail in the Advanced Implementation and Beyond course.

.....

StoneGate Protocol Agents can examine the data payload and also modify it. Therefore, when the source address is included in a packet's data, StoneGate can not only translate the original source address, but it can also translate the address embedded in the data.

Summary

In this chapter, you have read an overview of StoneGate's network address translation implementation. You have seen how StoneGate provides for several different translation methods, including:

- static source translation
- dynamic source translation
- destination translation
- destination port translation (PAT).

An example of the most common NAT types and their use in StoneGate, as well as a brief overview of proxy ARP provided additional information.

Review Questions

- What are the three main types of network address translation available?
- For what is proxy ARP used, and how is it applied in StoneGate?
- Besides address translation, what else can be translated for destinations, and why might it be used?

StoneGate User Authentication

Dr. Livingstone, I presume?

– Henry Stanley

This chapter introduces the principle of user authentication, as handled by StoneGate firewalls. The material will cover the basic StoneGate authentication (and authorization) principles, and give an overview of *Lightweight Directory Access Protocol* (LDAP) directories, supported authentication services, and the process of implementing basic user authentication.

Objectives

Upon completing this unit, you should be able to:

- explain the difference between authentication and authorization
- list the three authentication methods
- describe the implementation of user directories
- summarize the role of authentication services in StoneGate
- list the supported back-end protocols and services for authentication.

Introduction to Authentication

Authentication is the verification that someone or something is who or what they claim to be. It should be distinguished from *authorization*, which determines if someone or something is permitted to perform an action or access, after they have passed authentication.

Authentication can be performed in several ways. It is a generally accepted security principle that several methods be combined to increase the difficulty of compromising the authentication systems in place. For *user authentication*, we are concerned with the valid identity of a user, or person, so we will speak in terms of people, although the same principles apply to objects as well. With authentication, a person can prove their identity by one of the following means:

- something they are
- something they know
- something they have.

Each of these can be combined with one or more of the other two to strengthen the verification of the user. The most common authentication type is something the user knows, such as a password, or something they have, such as an authorization card.

Something They Are

To prove identity by something they *are*, the user presents a feature unique to their biology. In such *biometric systems*, the identity is determined by attributes such as voice, fingerprint, handprint, retina, signature or typing patterns.

Currently, the biometric systems are generally cost-prohibitive. They often require special equipment, such as fingerprint readers, retina scanners, high-quality microphones or other hardware, plus special software, and are therefore not practical for remote authentication over the Internet. Voice identification still requires intensive computational power to analyze and interpret the voice in question. Voice systems also typically require the user to speak under the same conditions, so if a person's voice is changed due to a flu, for example, difficulties in identification may rise.

Until this kind of technology evolves and becomes more commonplace, biometric systems are not a viable means for authenticating users, especially for remote Internet access to services.

Something They Know

Authentication with something they *know* is most commonly performed through some form of password authentication. That is, the users identify themselves by presenting a secret word or phrase. This system relies on the known thing being both secret and hard to guess. People, however, tend to be rather bad at creating unguessable passwords, and sometimes they are not very good at keeping secrets, either. If a password is hard to remember, or too long, the user may end up writing it down, moving the authentication principle to “something they have”.

The other problems with passwords for authenticating remotely over the Internet is that they can be captured in transit. An attacker can obtain the password as it travels from the remote machine to the site where the user wishes to authenticate. Although passwords can be encrypted, or hashed, the attacker can present the hashed password as well, without having to know the actual password used.

To make passwords a more secure means of authenticating remote users, two options are available. The first option is to use an *encrypted timestamp* with the password. This method ensures that the password was sent by the valid user in response to the system’s password request. However, encrypted timestamps require two things. First, there must be special client software that knows how to apply the encrypted timestamp to the password, and second, the time on both the client and server must be synchronized.

The other means of making traditional passwords secure involves using a challenge-response system. In such a system, the password you would use is in response to a particular prompt the server gives you.

Challenge-response systems may therefore seem even more complicated and impractical, since you would need to know not only one password, but a large number of passwords, one for each possible response.

To simplify challenge-response systems, you can instead memorize a rule for converting the challenge into a response. In this situation, you would know, for example, to reverse the third word of the challenge, uppercase it, shift the fourth character five more characters in alphabetical order, and so on.

Something They Have

A third, and perhaps the most successful means for authenticating users, is to use something they *have*. In such a system, the user possesses something unique, such as a list of one-time passwords, or an electronic card that is used for authenticating the user. Since this system relies on something in the user's possession, it is vulnerable to being lost, stolen, or destroyed.

Often, the “something they have” approach is combined with “something they know”. For example, many types of electronic cards generate one-time passwords, but require an identification number to be entered (something they know) to obtain the current, correct password. The popular SecurID system provides such a card, where a personal identification number (PIN) must be entered to obtain the password which is currently in synchronization with the password known on the server. This combination of something they have and something they know is often referred to as a *two-factor authentication system*.

StoneGate User Authentication

With StoneGate, users can be authenticated by both something they know (passwords), and/or something they have (electronic cards, one-

time password lists). The fundamental technology of StoneGate authentication uses the *Lightweight Directory Access Protocol* (LDAP). Third party authentication services, public key infrastructure (PKI) systems, and the RSA SecurID card are also supported.

In this section we'll examine the role of LDAP in StoneGate's user authentication, and how it integrates with the other components of the StoneGate Architecture.

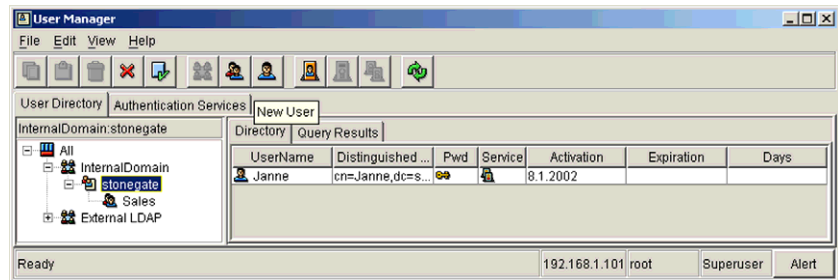
LDAP Directories

The LDAP directory service was developed by the University of Michigan. It is a lightweight version of the standard X.509 directory structure. Information about users and network devices, such as name, location, or company is stored in a database, which can be queried for information. LDAP is based on a client/server model, with LDAP servers storing the information, and LDAP clients that send and receive information from the server.

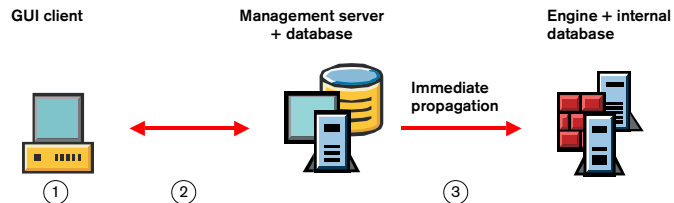
StoneGate internal database

StoneGate user management system provides an integrated database, but the software also supports the use of existing directory servers in a company. StoneGate Management Server handles the internal database in which all the user and group definitions are stored. Each firewall node stores a replica of this database, and any changes to the main database are propagated immediately to the nodes. This location provides increased performance and security for the firewalls, as they can access their local directories without communicating user information over the network.

Figure 9.1 exemplifies StoneGate system deployed with the built-in LDAP server. The internal LDAP directory is stored in the database of the Management Server and the directory is replicated to the firewall engines.

ILLUSTRATION 9.1 *Users and Groups are created and updated in GUI*

1. Users and groups are created and updated in GUI as in Illustration 9.1.
2. GUI communicates the data to the internal database of the Management Server.
3. Any updates to user data is propagated immediately to the internal databases of each node.

FIGURE 9.1 *Internal Management Server database*

Note: This solution is strongly recommended when the company has no external LDAP server and has no need for services of that type. This solution provides the most secure and reliable system as there is no need to authorize an external connection on the nodes.

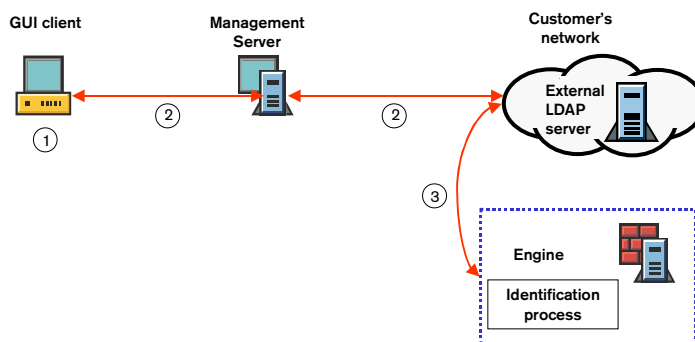
External LDAP server

An alternate approach can be used by enterprises that have an existing LDAP directory in place. In this event, the StoneGate system uses an LDAP client as part of the management system, which queries the corporate directory for user information. In order to use an existing directory, the administrator must add certain StoneGate specific attributes, or properties, to the user information. These include the StoneGate user name, password, and authentication services. The use of an external LDAP is covered in more detail in the *StoneGate Advanced Implementation and Beyond* course.

In StoneGate, the LDAP directory serves as the repository for all authentication decisions. The LDAP server can authenticate the user, or a third-party authentication service can be specified. In either case, the basic process of authentication by StoneGate is the same. Figure 9.2 illustrates the integration of an existing LDAP directory with StoneGate.

1. Users and groups are created and updated in GUI (User Manager).
2. The user information is passed by the Management Server to the external LDAP server.
3. StoneGate firewall engine enquires the external LDAP server for user identification information.

FIGURE 9.2 Existing LDAP integrated into StoneGate





.....

Caution: Deployment of an external LDAP solution with StoneGate may have some security considerations, as the whole system relies on an external component that is not part of the StoneGate system.

.....

How Authentication Is Performed

In StoneGate, user authentication is performed when remote users request access to services protected by the StoneGate firewall engines. The types of authentication permitted are defined in the access rule base as part of the security policy. Authentication rules mean that StoneGate then checks the configured LDAP directory to determine the authentication. First, the directory is queried to see if the user exists, and if any particular groups the user belongs to might also affect the authentication. If not, the authentication doesn't match, and the attempt is dropped. If the user is in the directory, the attributes are then checked to see which type of authentication should be performed. A "simple password" service means that StoneGate should verify the password supplied with the password information stored in the LDAP server. If another type of authentication is specified, StoneGate contacts the server providing that authentication service by forwarding the information to authenticate the user.

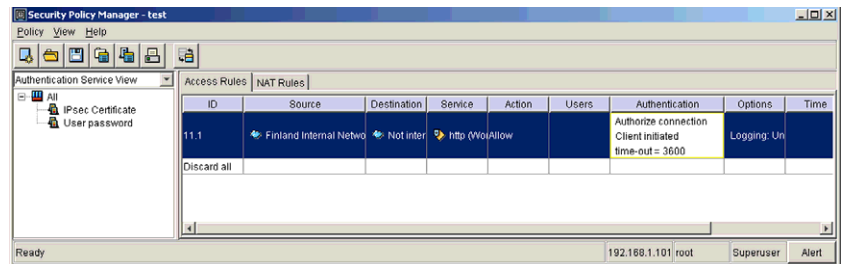
Authentication Types

There are two primary types of user authentication. The first type, *in-band authentication*, uses an existing connection protocol, such as telnet or FTP, and authenticates the user with the firewall engine as part of that protocol's authentication exchange. StoneGate, however, supports *out-of-band authentication*, which uses a separate connection for authentication, in addition to the protocol connection for the service being used. Out-of-band authentication enables support for a larger

variety of protocols, without having to modify any of the existing protocols in any fundamental way.

Out-of-band authentication can be further categorized on the basis of the origin of the authentication request. Requests for a user to authenticate themselves can be initiated by either the firewall engine, or by the client software on the remote system. Illustration 9.2 shows an example authentication rule located in the Security Policy Manager.

ILLUSTRATION 9.2 *Authentication Field in the Access Rules*



Firewall-Initiated Authentication

For firewall initiated authentication, the firewall establishes a connection to the remote system, and the client software then prompts the user for their ID and password. The request is then sent to the appropriate authentication service and evaluated. Which type of authentication service to use is specified in the corresponding access rule in the security policy. Two types of authorization can be performed by firewall initiated authentication:

- client IP authorization
- current connection authorization.

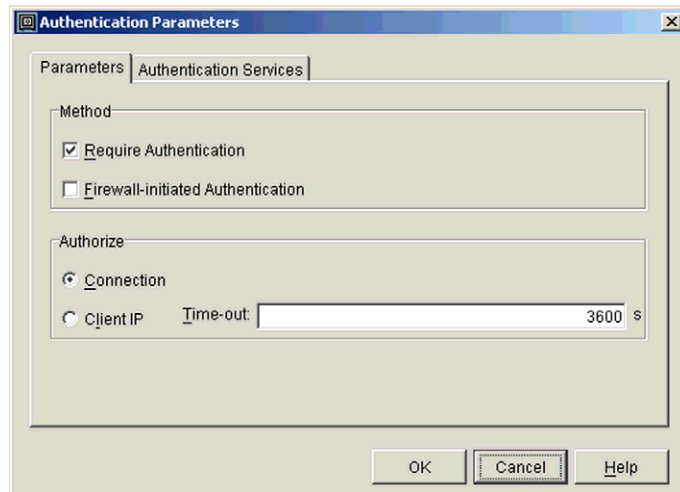
ILLUSTRATION 9.3 *Authentication Parameters window*

Illustration 9.3 shows the Authentication Parameters in the Security Policy Manager.

With *client IP authorization*, the IP address of the remote system is granted access to the destination for a specific period of time. This type of authorization is useful when a remote system needs to establish any number of connections, for any specific period of time. For example, a user authenticated by this method may be authorized to access the corporate intranet, mail system, and file shares for an hour. The disadvantage to client IP authorization is that all users of that remote system will have the ability to contact the authorized services without having to perform their own authentication.

Client IP authorization is also supported by client initiated authentication.



.....

Caution: Unix systems, because they are typically multi-user systems, should not perform client IP authorization, since every user on that system will have access through one user's authentication.

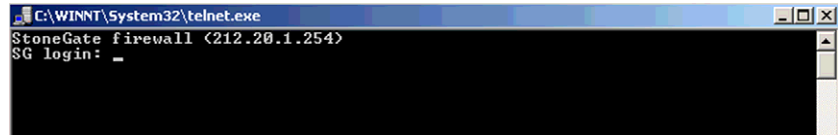
.....

Current connection authorization, on the other hand, is granted through firewall initiated authentication for the current connection request only. A user may attempt to access a particular service through the firewall, such as the corporate mail server, for example. The firewall will initiate an authentication request with the remote system, prompting the user for an ID and password. If the authentication service specified accepts the user, access is granted for that connection. Thus, the user will be allowed to connect to the mail server that one time, but to no other service without additional authentication.

Client-Initiated Authentication

Client-initiated authentication is commonly used in situations where it may not be possible to establish a connection from the firewall to the remote system. For example, a user requesting a service may be located behind one or more additional firewalls. In such situations, the client software of the remote system initiates a connection with the firewall protecting the service requested. Two types of authorization are supported by client initiated authentication:

- client IP authorization
- upcoming connection authorization.

ILLUSTRATION 9.4 *Example of Authorization request for login via Telnet*

As with firewall initiated authentication, *client IP authorization* is supported by client initiated authentication. The client software of the remote system initiates a connection with the firewall, and, upon successful authentication of the user, that remote system is granted access to all authorized services behind the firewall. Again, caution should be taken with this approach, as any users on the remote system will all have access based on one authentication.

The second type of authorization supported by client initiated authentication is *upcoming connection authorization*. With this type of authorization, the client software of the remote system initiates an authentication request to the firewall. The firewall, upon completing a successful match and authentication service in the security policy, grants access for the next connection from that host. Although more secure than client IP authorization, this approach is also potentially vulnerable with multi-user systems.

Authentication Methods

The StoneGate system supports many third-party authentication services, in addition to performing simple password authentication based on information in the internal database. In this section, we'll provide a brief overview of the authentication methods supported. A more thorough account of external authentication is presented in the *StoneGate Advanced Implementation and Beyond* course.

Internal Authentication

As discussed before, the internal database of StoneGate stores usernames and passwords, which can always be used for simple password authentication. When “**User Password**” has been specified as the authentication service for a user, the users’ username-password pair is checked from the internal database whenever a rule requires authentication for a connection.

External Authentication

Simple password authentication can also be deployed when an external LDAP server is configured for authentication. The use and implementation of an external LDAP solution is covered in the *StoneGate Advanced Implementation and Beyond* course.

There are also many third party authentication services available on the market today. Many of the most common authentication servers, such as RSA ACE/server, are supported by StoneGate, enabling an administrator to use existing servers, or to choose from a variety of products to incorporate into the StoneGate system. Authentication methods, such as one-time passwords and token cards, are supported by standard RADIUS and TACACS+ back-end protocols, which are presented here briefly. The external authentication parameters for StoneGate can be set in the User Manager.

RADIUS

Remote Authentication Dial-In User Service – known as RADIUS – was originally developed to provide authentication for dial-in terminal servers or other devices that support remote access over phone lines. Today, RADIUS has been adapted to support other forms of remote access authentication, such as TCP/IP connections over the Internet.

For StoneGate, authentication is performed when a remote user contacts the StoneGate firewall engines, requesting access to some

service protected by the firewalls. If the authentication service defined in the security policy or LDAP server is “RADIUS”, the firewall engine uses its integrated RADIUS client software to contact a specified RADIUS server. More than one RADIUS server can be configured, ensuring availability in the event that the primary RADIUS server cannot be reached. As part of StoneGate’s distributed architecture, the RADIUS server can be located anywhere on the network, or across the Internet.

RADIUS client/server communications encrypt the exchange of password communications, but not other data. Because RADIUS servers perform auditing as well, usernames and other information are also transmitted, but they are not encrypted.

TACACS+

Based on TACACS, which may – or may not – be an acronym for Terminal Access Controller Access Control System (the origins of this acronym have been lost), TACACS+ enables a more secure method of dial-up user authentication than RADIUS.

For example, TACACS+ includes encryption support, using a shared key between the client and server. It can also supply additional IP address information for remote clients, assigning them configuration information. User information, such as IDs and passwords, are transmitted in an encrypted form, using MD5. Because of these features, TACACS+ is a significant improvement on TACACS or RADIUS.

Summary

In this chapter, we have covered basic authentication principles, and the design of user authentication in StoneGate. LDAP, and how it fits into StoneGate’s distributed architecture, as well as the role it plays in authentication services was also covered. Third party authentication

services and how they are defined and enabled in StoneGate were reviewed briefly.

Review Questions

- What is the difference between authentication and authorization?
- Where can StoneGate store user data?
- Why is it safer to base the user authentication on the StoneGate internal database instead of using an external LDAP server?
- How does StoneGate provide support to third party authentication services?

VPN Fundamentals

Entropy is a figure of speech...a metaphor. It connects the world of thermodynamics to the world of information flow.

– Thomas Pynchon

This chapter gives a basic introduction to virtual private networks (VPN). It will give some fundamental information about those different cryptographic technologies that are relevant to the way StoneGate implements VPNs. It also outlines how VPNs are actually created with StoneGate.

Objectives

Upon completing this unit, you should be able to:

- list three benefits of using a VPN
- understand the basics of cryptography
- understand the two main protocols of the IPsec protocol suite
- describe the general IKE negotiation process
- explain the use of different authentication methods
- describe how VPNs are implemented in StoneGate.

Introduction to VPN

Basically, a virtual private network (VPN) is a private network that is extended over public networks by using cryptographic protection. Modern, distributed organizations increasingly use the Internet for

connections between different sites, and between mobile clients and the corporate LAN. The use of VPNs can be far less expensive than the use of leased lines. In addition, they offer good flexibility, as a wide range of different access media can be used to establish VPN connections over the Internet; these include:

- dial-up connections
- ISDN
- xDSL
- Frame Relay
- E1 and T1 lines.

You might consider using VPNs to provide secure connections to your LAN for:

- branch offices
- supplier or partner offices
- customer offices
- telecommuters
- mobile workers.

With the use of cryptography, the integrity and confidentiality of the data can be protected when transmitted over an untrusted public network. All the traffic sent through a VPN is encrypted at the transmitting end and decrypted at the receiving end. The identity of communicating parties is checked with user authentication based on public key exchange and certificates or with a shared secret key.

In IP based VPN solutions, traffic is commonly protected by *IPsec protocols*. IPsec doesn't affect the routing of IP packets in the public network, therefore protected VPN traffic is handled as efficiently as normal, unencrypted IP based traffic. The network elements that use IPsec encryption protocols to negotiate and establish secure, virtual tunnels between the connected sites are referred to as *security gateways*.

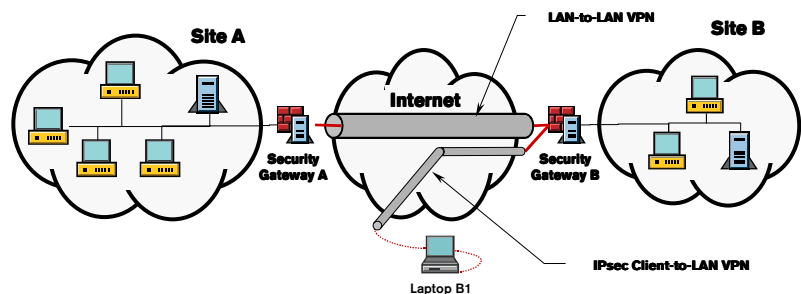
A security gateway can be a firewall, a router, or any other VPN compatible gateway device. A *site* consists of a security gateway and any number of network elements connected to it.

StoneGate provides several advanced VPN features. As a *security gateway* it can be used to establish VPN connections between different sites. It supports both LAN-to-LAN and client-to-LAN connections (see Figure 10.1). In addition, StoneGate supports also Multi-Link VPN connections that introduce high availability to the VPN tunnels. StoneGate's clustering technology means also that security gateways can consist of more than one node, all working as one virtual entity. Multi-Link VPN is covered in more detail in the *StoneGate Advanced Implementation and Beyond* course.



All the VPN related settings can be easily managed with the VPN Manager. With it, StoneGate administrators can implement highly secure, encrypted and authenticated VPN connections. The VPN settings can be adjusted for individual VPNs, tunnels and connections, which provides a high degree of granularity.

FIGURE 10.1 VPN architectures supported by StoneGate



However, before taking a closer look how VPNs are implemented with StoneGate, it's worthwhile to get an overview of the different underlying technologies.

Overview of Cryptography

Encryption is a method of protecting information by making it comprehensible to authorized individuals only. This is done by taking the original information, known as *plaintext*, and running it through a mathematical *algorithm* with a *key*—ideally a large random number—to create a secret message known as *ciphertext*. The ciphertext is meaningless until it is processed again, i.e. decrypted, with the same mathematical algorithm, using the same or—in case of modern asymmetric cryptography—a related key, in order to return it to the original plaintext state.

Cryptography has a long history, and it has been used for concealing messages for centuries; the earliest known encrypted texts are ca. 4,000 years old. Modern cryptography really started to evolve in the 20th century, one of the driving forces being the need to conceal military secrets. It can be said that scientific secret key cryptography is founded on C.E. Shannon's theories, published after World War II. The increased academic interest and development during the last two or three decades has expanded the domain of cryptography from seclusion in the military agencies to the public domain.

In the computer age, the importance of encryption has increased immensely. With the swiftly growing computing power, there are more and more practical applications for strong cryptographic solutions. Cryptography has become a cornerstone of secure digital communication, and it is widely used in all kinds of services, for example:

- digital financial transactions
- mobile telephony

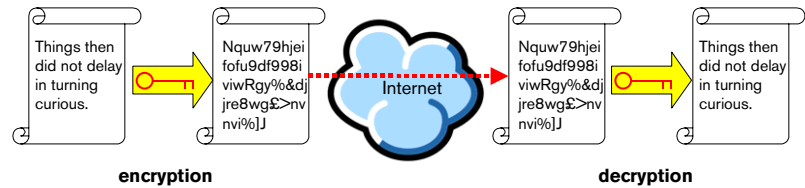
-
- digital signatures
 - virtual private networks.

Unfortunately, the processing power can and is used also for undermining these services which are relying on encrypted communication. Therefore it is crucial that the algorithms and keys used to protect data are tested and proven to be safe. Only well-known and safe algorithms and non-predictable keys should be used. Ideally, true sources of randomness should be used for generating keys. If pseudo-random number generators are used, they must provide enough entropy to make predicting the generated keys as difficult as possible.

The two distinct types of key-based encryption—symmetric and asymmetric—are presented next.

Symmetric Encryption

Symmetric encryption, also known as *secret key encryption*, uses a single shared key for encrypting and decrypting information. The plaintext is encrypted by a key and the resulting ciphertext can then be stored or sent to a recipient—for example, over the Internet. At the receiving end, the message is decrypted with the exact same key that was used for encrypting it (see Figure 10.2). Methods based on secret keys have been in use for thousands of years. The crucial issue for the security of symmetric encryption is the strength of the key itself. With modern computing resources, a poorly defined key can be cracked in a matter of seconds. Basically, it is recommended that the secret keys be created by good random number generators. The algorithms are usually well-known and documented (Triple DES, IDEA, Blowfish, etc.).

FIGURE 10.2 *Basic principle of symmetric encryption*

However, no matter how good the actual key is, the use of a common key poses a dilemma: how to share the secret key between the parties, so that it's not revealed to anyone unauthorized? In modern cryptographic systems, the shared secret key is valid only for one session, hence it's also called the *session key*. This has the benefit that if the key were compromised, it couldn't be used for decrypting any other messages.

One viable solution to the key distribution problem is the use of key agreement algorithms. The secure management of keys can be achieved by the use of, for example, Diffie-Hellman key agreement method. There, the actual session key never needs to be exchanged; instead, both parties can compute the secret key independently from the generated keying material. For more information, see section *"Diffie-Hellman Key Agreement"* on page 157.

Several symmetric algorithms are in use today, and new ones are being developed as older ones eventually become unreliable. They are used mainly for encrypting the bulk data. The asymmetric solutions presented next, on the other hand, are much slower and thus aren't suitable for encrypting big amounts of data. Instead, they are used for several other services.

Asymmetric Encryption

Asymmetric encryption, also known as *public key encryption*, uses two separate keys for encrypting and decrypting information. Public key solutions only started emerging in the 1970s, but they have become widely used in digital communication systems.

In asymmetric cryptography, each communicating party generates a unique pair of keys in a predefined way. The *public* key is available to anyone who needs to use encrypted communications, and it can, e.g., be published on the Internet. The *private* key is kept confidential and should be known by the owner alone. The two keys are mathematically related, but basically one can't be used for resolving the other. The keys used in asymmetric algorithms (such as RSA or DSA) are usually much longer than those in symmetric encryption. There are several types of public key algorithms, some more suitable than others for specific kind of services.

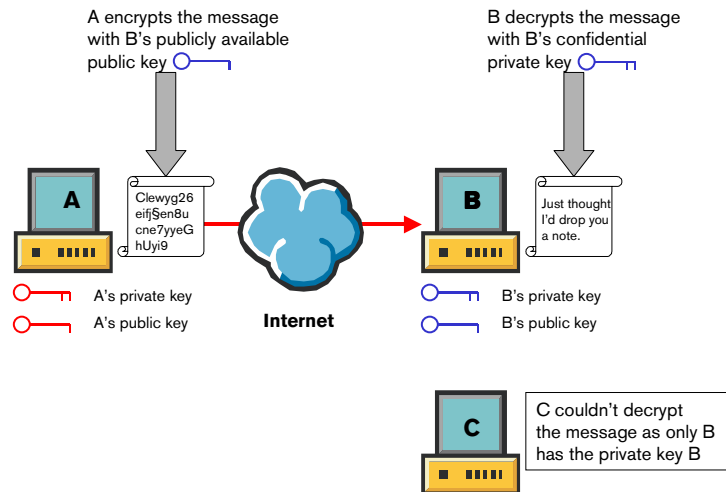
Typically, public key encryption is not used for encrypting bulk data because symmetric encryption is much faster and less resource intensive for that purpose. Instead, it is generally used for encrypting and authenticating the shared key used in symmetric encryption; this type of combination of symmetric and asymmetric methods is sometimes called *hybrid encryption*. It can also be used in creating digital signatures; see section “*Digital Signatures*” on page 159.

In general, public key algorithms can be used for following purposes:

- *encrypting and decrypting*: the sender encrypts the message with the recipient's public key. Only the recipient can decrypt the message, being the only one who possesses the corresponding private key. As noted before, asymmetric encryption is generally not used for bulk data, but rather, e.g., for encrypting shared symmetric keys. The basic principle of public key cryptography is presented in Figure 10.3.

- *digital signatures*: the sender can encrypt a message digest computed from a message with the private key. The recipient(s) can verify the sender's identity by decrypting the message digest with the sender's public key. The message digest can only be decrypted correctly if it really was encrypted with the corresponding private key. For more information, see section “*Digital Signatures*” on page 159.
- *key exchange*: the communicating parties can exchange keying material to generate a shared session key without risking the actual key. Using the exchanged public keys and their confidential private keys for computing, both parties should independently get the same result which can then be used as the shared key. One of the most widely used key exchange methods is *Diffie-Hellman*. For more information, see section “*Diffie-Hellman Key Agreement*” on page 157.

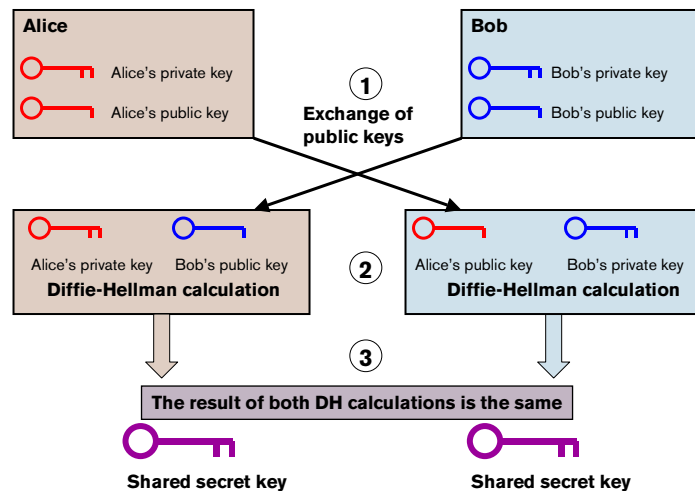
FIGURE 10.3 Basic principle of asymmetric encryption



Diffie-Hellman Key Agreement

Diffie-Hellman is a classic public key agreement scheme used to generate a secret key between the parties without actually communicating any secret information. This way, crucial keying information that could be captured by eavesdroppers doesn't need to be sent over insecure channels. Each party generates a pair of keys: a public one and a private one. These keys are mathematically related to each other. Only the *public* parts of the Diffie-Hellman key are exchanged, and the whole secret key is calculated independently by both parties with the use of their *private* key. The resulting key is common to both, and it can be used as the shared secret key for the communication between the parties using symmetric encryption.

FIGURE 10.4 Diffie-Hellman key agreement scheme

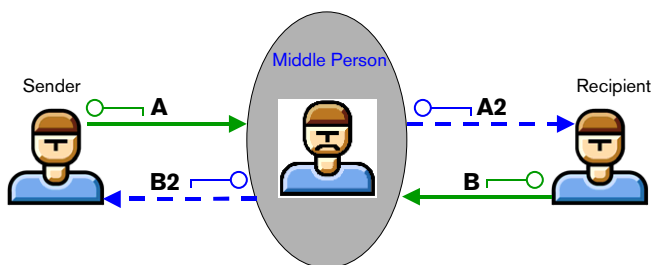


Authentication and Integrity

The problem with using encryption for establishing a VPN, however, is that at some point you have to begin exchanging key information for the first time. Obviously, you want to know that the entity with whom you are exchanging information is, in fact, the correct party. Also, you want to ensure that the information you are sending and receiving is not altered or corrupted in some way during its journey. This authentication and integrity assurance is accomplished through the use of digital signatures and digital certificates. These components are used increasingly for fulfilling the requirement of authentication and data integrity in, for example, e-commerce. Together with public key cryptography and key management, they form a comprehensive digital security system known as the *Public Key Infrastructure (PKI)*.

For example, the Diffie-Hellman key agreement method described above, while a useful method for generating secret keys, doesn't by default guarantee the authenticity of the communicating parties. Therefore, it is vulnerable to a *man-in-the-middle attack* (see Figure 10.5). In other words, a third, unauthorized party may be in position to intercept the sender's public key "A" and send a bogus public key "A2" to the recipient instead. Likewise, when the recipient replies by sending the public key "B", the malicious "middle person" replaces it with yet another fake key "B2" which the original sender will take for the recipient's key. This way, the middle person can establish two secured connections, one with A and another with B, while they think they are communicating directly with each other. In fact, the hijacker is able to meddle with the communication between A and B freely, decrypting their messages and possibly altering or replacing them.

This example should emphasize the need for a secure way of authenticating the communicating parties and protecting the integrity of the sent data. The following sections will describe how this risk can be overcome by the use of modern digital authentication methods.

FIGURE 10.5 *Man-in-the-middle attack during Diffie-Hellman exchange*

Digital Signatures

Digital signatures constitute a means to achieve both sender *authentication* and data *integrity*. They certify that the message has in fact been sent by the alleged sender, and that the data has not been tampered with after it has been signed. They also support *non-repudiation*, which means that digital signatures can, for example, be used as evidence for identifying the sender and the timestamp of a message. Digital signatures rely on public key schemes.

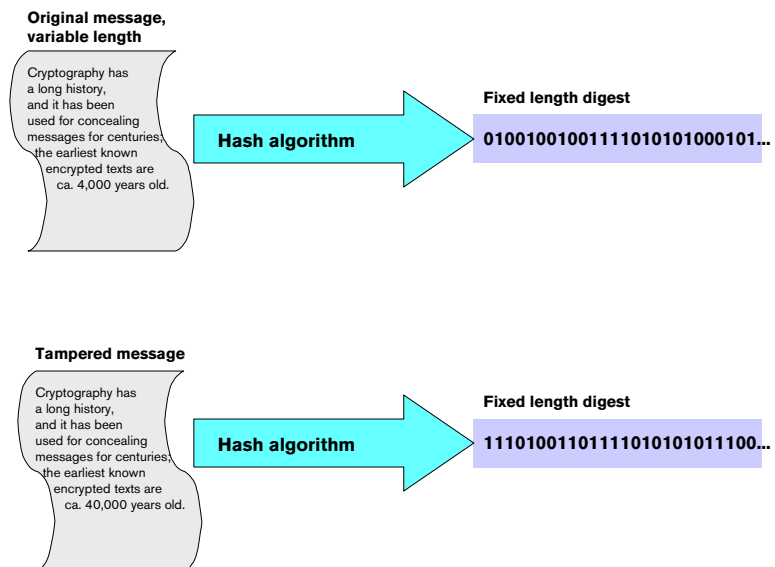
The signature is created with the sender's private key, and validated by the recipient(s) with the sender's publicly known key. The actual signature is typically generated by encrypting the *hash* (i.e., message digest) of the message with the sender's private key. A received digital signature is verified by decrypting it with the sender's public signature key, and matching the result against the hash of the original document.

Message Digests

Message digests, also known as one-way *hash functions*, are algorithms that are used to compute a fixed length “fingerprint” of a given message. The algorithm doesn't require a specific key for contracting the original message. These functions are one-way only; so the original

message cannot be recomputed from the digest. Message digests are difficult to counterfeit as each message has with a very high probability a unique hash. Any tampering of the message would inevitably alter the message digest and thereby reveal that the integrity of the message has been violated. Because of these characteristics, the message digests are used for digital signatures and for ensuring the integrity of data during transit. Figure 10.6 exemplifies how a fixed length, unique hash results from applying the hash algorithm to a message. Observe also how changing the original information will inevitably change the digest as well. Consider the implications if the numbers here were, e.g., financial figures or medical data.

FIGURE 10.6 *Message digest*

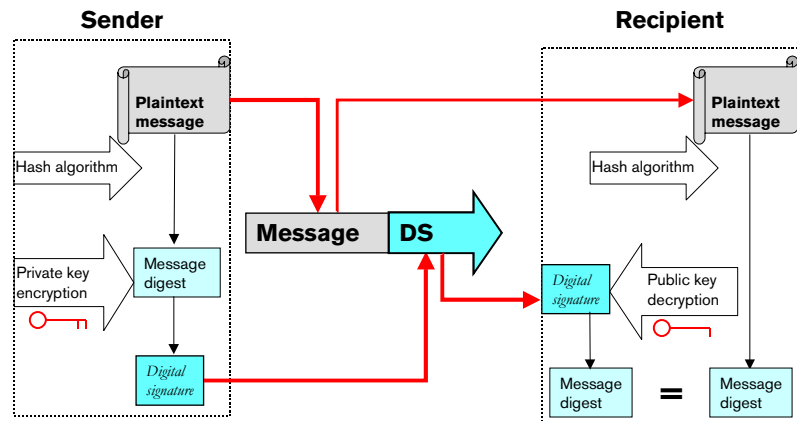


In Figure 10.7 we can see how message digests and public key encryption are used in combination to produce a secure digital signature. A message digest is contracted from the original plaintext message. The sender then encrypts this digest with the private key and

attaches the resulting digital signature to the plaintext message. The recipient verifies that the sender really is who he or she claims to be by decrypting the digital signature with the sender's publicly known key. As we remember, data encrypted with a person's private key can only be decrypted with the corresponding public key.

In addition, the recipient uses the same hash algorithm to calculate a digest from the received plaintext message. If the message hasn't been tampered with during transit, the result should be identical to the original digest, because identical messages should produce identical digests. The recipient can now compare the digest from the message itself to the decrypted digest of the digital signature. These two message digests should prove to be identical; otherwise the authenticity and integrity of the exchange have been compromised.

FIGURE 10.7 *Digital signature*



Digital Certificates

Digital certificates are yet another authentication method employing digital signatures and public key cryptography. The certificates are

created by trusted third parties known as *Certificate Authorities* (CA). They create the certificates by signing the public keys and identity information of the communicating parties with their own private keys. This way, it's possible to verify that a public key really belongs to the alleged party. A public-key certificate identifies the owner of a public key. Typically, certificates include the following information:

- name and identification of its holder
- CA name
- public key
- certification class, and
- validity time.

Each communicating party may be required to present its own certificate signed by a trusted CA verifying the ownership of the corresponding private key. Additionally, the communicating parties need to have a copy of the CA's public key to be able to trust the CA and validate certificate signatures issued from that CA.

The CAs are responsible also for providing the service of *certificate revocation lists* (CRLs). The CRLs include the certificates that are reported as no longer valid because, for example, the corresponding private key is lost or stolen or the identity information has changed.

IPsec Overview

As previously explained, the most widely used standard for implementing VPNs is the *IPsec protocol set* (as defined in RFC 2401). It specifies the use of encryption and/or authentication methods for secured IP networking. StoneGate's VPN solution is based on this standard. In this section, we will first give an overview of the basic protocols used in IPsec. Then we will explain how VPN tunnels and connections are established through Internet Key Exchange (IKE) phases 1 and 2.

The IPsec protocol suite is based on two main protocols: the *Authentication Header (AH)* protocol and the *Encapsulating Security Payload (ESP)* protocol. These protocols provide several network security services for implementing the VPN encryption policy at the IP traffic level. The communicating parties negotiate to establish tunnels – two unidirectional connections called *Security Associations (SA)* – before they can send secured data. All the required information (such as keys, algorithms, modes, and lifetimes) for the support of the IPsec connection between two sites is recorded in the SA data. These SAs are implemented by means of AH and ESP protocols, and their contents are negotiated during the Internet Key Exchange (IKE) phases (see “*VPN Policy Parameters*” on page 175). The SAs are stored by both of the communicating devices in a *Security Association Databases (SAD)*, and they are identified with a *Security Parameter Index (SPI)* value which is included in the AH and ESP protocol headers.

Authentication Header (AH)

AH is basically an authenticating protocol that guards packet integrity and authenticates the identity of the source host. It can additionally provide protection against replay attacks, but it does not provide data encryption. The use of AH is relevant for performance issues if no encryption is needed, or for legal reasons if ESP encryption algorithms are forbidden by local law.

The protocol inserts an authentication header into the original IP packet. The IP header is modified to indicate that the packet includes now an authentication header. The latter is composed of several fields for specifying the transport-layer protocol in use, for identifying the appropriate SA for the packet with SPI, for setting a sequence number to prevent replay types of attacks, and for authenticating the packet with a digital signature.

To support the data integrity service, AH computes a keyed hash signature (with MD5 or SHA-1 algorithm) based on the data payload

and several parts of the IP header and authentication header. This enables the gateway to verify that packets and their corresponding IP headers are coming from a trusted security gateway or host.

Encapsulating Security Payload (ESP)

ESP is both an authenticating and encrypting protocol that provides data integrity and encryption services. It additionally provides authentication services and protection against replay attacks.

The protocol inserts an ESP header to the original IP packet. The packet is encrypted applying the encryption algorithm specified in the negotiated SA from the IPsec proposal. The ESP header is composed of two fields; one for identifying the appropriate SA for the packet with SPI and one for setting a sequence number used to prevent replay attacks. Additionally, the ESP protocol includes some padding data in the trailer. ESP padding is a feature specifically used to conceal the actual length of the packet payload.

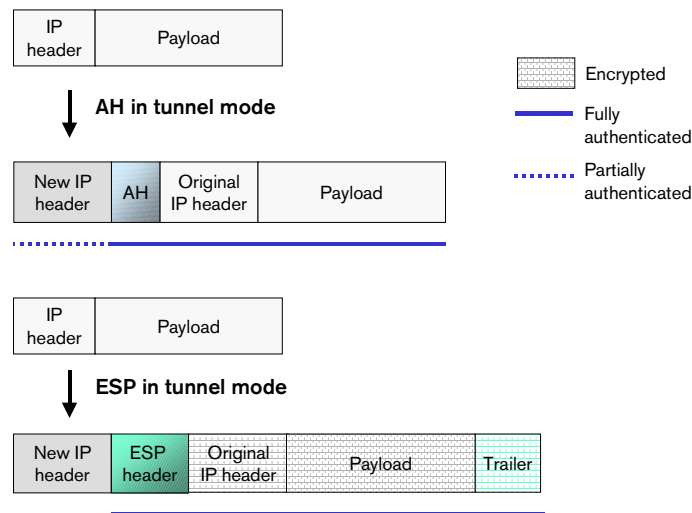
To allow the checking of data integrity, ESP authentication material is inserted unencrypted at the end of the packet, after the trailer. ESP can compute a hash signature (MD5 or SHA-1) based on the payload and the original and ESP headers.

Tunnel Mode and Transport Mode

Two different modes are specified for encrypted traffic: tunnel and transport modes. StoneGate firewalls execute the AH and ESP protocols via *tunnel mode*. The tunnel mode consists in encapsulating the original packet into a new one as shown in Figure 10.8. With ESP, it conceals the original source and destination addresses of IP headers and only discloses the addresses of the communicating security gateways. In contrast, the *transport mode* inserts the ESP and AH headers between the header and the upper layer protocol field of IP packets.

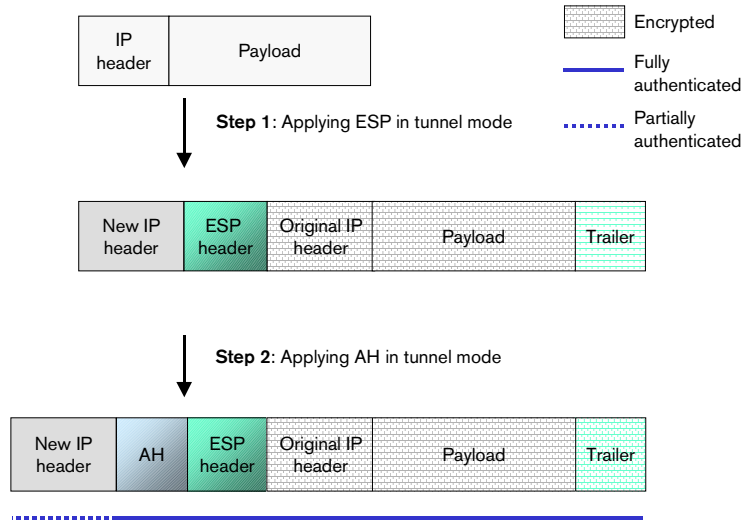
The tunnel mode is automatically applied by the firewall engine when controlling and protecting access to communications between security gateways. The original packet constitutes the payload of the ESP or AH headers. The encapsulated packet is encrypted and/or authenticated depending on which protocol is specified.

FIGURE 10.8 AH and ESP headers in tunnel mode



Combination of AH and ESP

The AH and ESP protocols can be combined to enhance data integrity (see Figure 10.9). Their combination provides the same level of confidentiality as ESP; authentication is performed with the help of the AH procedure. Typically, ESP is used to encrypt the data payload. The ESP protocol is applied to the packet before the AH protocol which can authenticate the entire data payload including the ESP header.

FIGURE 10.9 Headers in AH-ESP combination

Internet Key Exchange (IKE)

Authentication and encryption are closely interwoven concepts in VPN encryption technology. In order to establish secure communications, strong authentication and encryption measures must be negotiated between the communicating sites. The IPsec protocols supported by StoneGate rely on the *Internet Key Exchange* (IKE) negotiation procedure. During IKE negotiation, the required parameters, and key exchange and authentication methods for IPsec traffic are agreed upon and recorded in SAs. The SAs are implemented by using the AH and/or the ESP protocol. The initiator may make multiple proposals for IKE and IPsec SAs, but only one of these will be sustained. The different parameters are explained in more detail in section “*VPN Policy Parameters*” on page 175.

The IKE negotiation consists of two separate phases, which are described below.

IKE Phase 1 Negotiation

During this phase, a secure communication channel for further negotiations is established. The communicating parties are authenticated, and the authentication information is secured by generating encryption keys. The exchange of the keying material is achieved with Diffie-Hellman key agreement method. The authentication of the identities is verified with the help of public key methods or with pre-shared keys.

An *IKE SA* is created at each end of the connection. This SA contains information about the algorithms for encrypting and signing the data exchanged during the rest of this phase and the following phase. The IKE SAs used between security gateways form a bidirectional secured tunnel that is used to negotiate IPsec SAs during the next phase.

Two different modes are supported for this phase:

- **Main Mode:** It protects the identity of each communicating party. With this mode, the communicating parties exchange six packets.
 - The first exchange of packets negotiates an agreement for the SAs on the basis of IKE proposal;
 - the second exchange shares Diffie-Hellman public keys and some required ancillary data;
 - the third exchange authenticates the identities and all previously exchanged data, either with digital signatures or public key encryption algorithms. Certificates can also be used for the authentication.

- **Aggressive Mode:** It does not protect the identity of the communicating parties, but helps preventing denial of service attacks. With this mode, only three packets are exchanged in a more compact format.
 - The first two packets negotiate an agreement for the SAs, share Diffie-Hellman public keys with some required data, exchange unencrypted identities; and authenticate the remote party.
 - A second unidirectional exchange authenticates the initiator of the negotiation.



Caution: Because the aggressive mode does not provide identity protection, it should only be used with proper consideration.

FIGURE 10.10 IKE phase 1 main mode

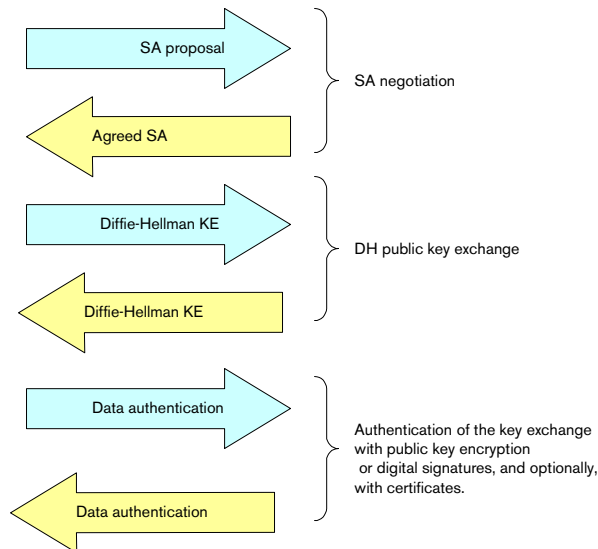
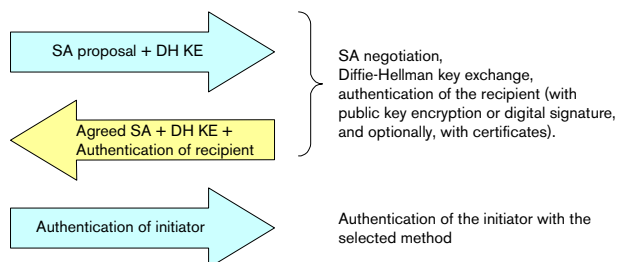


FIGURE 10.11 *IKE phase 1 aggressive mode*

IKE Phase 2 Negotiation

During this phase, also known as *Quick Mode*, a pair of *IPsec SAs* is negotiated using the SAs from the first phase. The encryption/decryption procedures applied to the data are agreed upon. The information for IP traffic protection is contained in the IPsec SAs. The negotiation of IPsec proposals is encrypted with the keys already agreed upon in the IKE SAs. The generated unidirectional IPsec connections are used for the actual encrypted data traffic between the security gateways. The SA proposal specified for this phase sets the lifetime of the IPsec SA.

The IPsec SAs can be negotiated at different granularities; for each pair of networks, hosts, protocols, or ports. However, negotiating SAs on a per protocol or per port basis will increase memory consumption significantly due to the high number of resulting SAs.

The phase 1 involves quite heavy computation, thus the IKE SAs are renegotiated far less frequently than the IPsec SAs. When an IPsec tunnel expires—depending on its lifetime parameters—only the phase 2 negotiation is performed again. It is renegotiated on the basis of the IPsec proposal parameters through the IKE tunnel. The renegotiation results in new keying material for the IPsec traffic. Fresh keying material can be created for each new negotiation of both IKE and

IPsec proposals with the help of Diffie-Hellman generated secret keys.

Manual IPsec Mode

As an alternative to IKE, manual IPsec key exchange is also supported by StoneGate. In the case of manual IPsec keying, manually entered authentication and encryption keys are used for encrypting/decrypting the traffic. As IKE is not used, the encryption key cannot be changed regularly, and protection against replay attacks cannot be ensured. Support for manual IPsec is provided mainly for compatibility with third-party products.



.....

Note: The administrators need to be fully aware the effects of using different IKE negotiation and IPsec mode parameters. A selection of parameters can have serious impact on the security and performance of the overall encryption policy.

.....

Building VPNs with StoneGate

This section will describe how StoneGate makes use of the techniques described earlier in this chapter to form secure VPN communications.

Security Gateways and Sites



When implementing a basic gateway-to-gateway VPN with StoneGate GUI, you will first define the *security gateway* nodes between which the VPN is to be established. It is possible to define both *internally* managed (i.e., StoneGate firewall) and *externally* managed (i.e., third party gateway or StoneGate outside the administration scope) security gateways in the VPN Manager. The end-points for the VPN connections must be defined at each gateway. They are the IP

addresses on the interface through which the gateway is connected to a network link.



.....

Note: StoneGate has two security gateway types: internal security gateways, and external security gateways. Internal gateways are all security gateways managed by that StoneGate management system, regardless of their location. External gateways are security gateways managed by another StoneGate management system, or a third-party firewall's management tools.

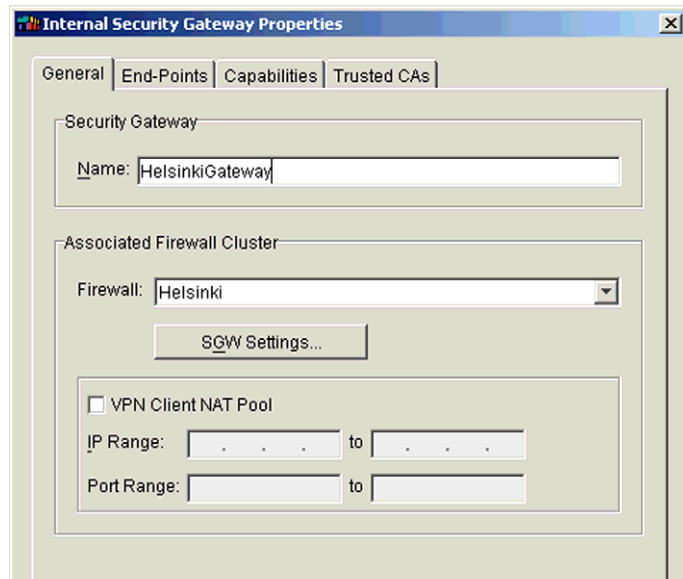
.....



.....

Note: Third-party gateways can also be configured with a dynamic IP address as the end-point address. In that case, alternative identity information must be defined. The options for the alternative identity are DNS name, e-mail address and distinguished name.

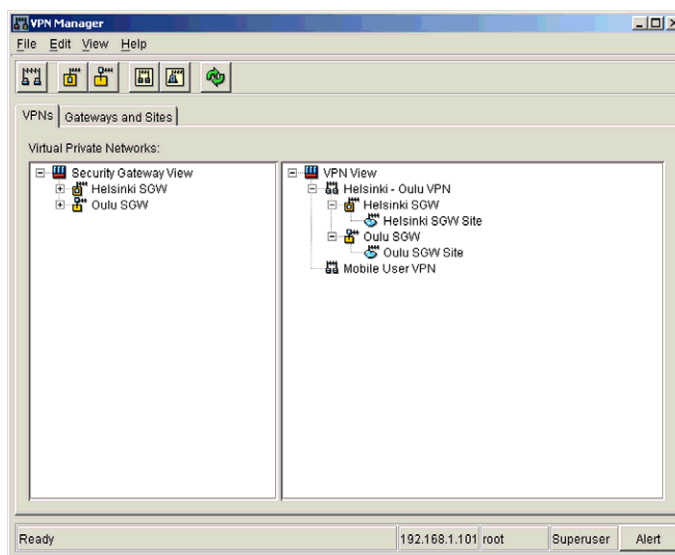
.....

ILLUSTRATION 10.1 *Internal Security Gateway Properties*

A number of *sites* is associated with each gateway. Each site can represent a group of network elements or a whole network connected to the security gateway. Several sites can be assigned to the same gateway. In the VPN Manager, sites can simply be moved under the appropriate gateways. Access to mobile VPN IPsec clients can be granted to specified sites. The features and use of StoneGate VPN Client will be covered in the *StoneGate Advanced Implementation and Beyond* course.

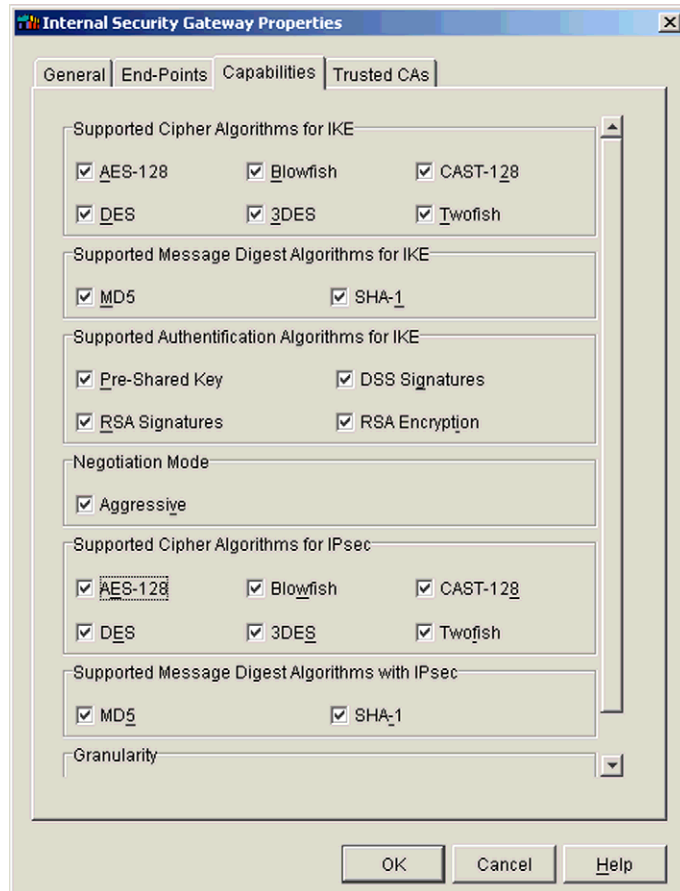


After defining the gateways and the associated sites, you can create a VPN element between gateways. This as well can be done in the VPN Manager by simply dragging the appropriate gateway elements under the same VPN element.

ILLUSTRATION 10.2 *Created Security Gateways and VPNs in VPN Manager*

After the VPN structure is thus set, the encryption and authentication measures must be defined. StoneGate supports the IPsec protocols for the implementation of VPN for IP traffic. A set of default settings is provided but you can adjust several encryption algorithms, authentication methods, and other parameters that you want to support at your gateway. The different parameters are described in section “*VPN Policy Parameters*” on page 175.

An encryption policy can be defined either globally for all corporate security gateways and VPN connections or for individual connections involving a pair of sites.

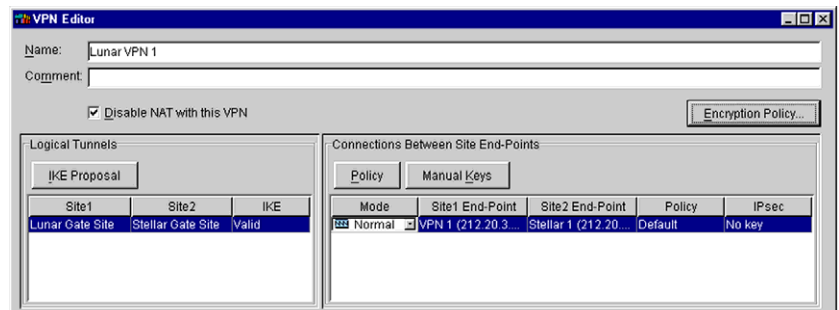
ILLUSTRATION 10.3 *Internal Security Gateway Properties; Capabilities tab*

Tunnels and Connections

Virtual *tunnels* are generated between each pair of sites, and they represent a set of secured connections. These bidirectional tunnels are created during the first negotiation phase of the VPN communications, i.e., during the Internet Key Exchange (IKE) phase 1. During the next phase, the actual unidirectional IPsec *connections* are generated between each link that connect both security gateways of

the VPN to the Internet. They represent combinations of the end-points for each pair of sites, and they are generated during the IKE phase 2. You can define all the required parameters for IKE in the VPN Manager. Illustration 10.4 exemplifies a basic VPN between two single gateways with one logical tunnel and one VPN connection.

ILLUSTRATION 10.4 *VPN properties window*



The operating mode of individual connections can be tuned between **Active**, **Standby** and **Disabled**. Furthermore, VPN traffic can also be load-balanced between multiple links, introducing high availability to VPN traffic. This Multi-Link VPN shall be discussed in detail during the *StoneGate Advanced Implementation and Beyond* course.

Next, we shall present the different parameters for authentication and encryption that you can set for VPN traffic in StoneGate.

VPN Policy Parameters

In this section we shall go through the main parameters that are related to defining of the general VPN policy in StoneGate. These parameters relate to the confidentiality, integrity, and authentication of data in the VPN environment. They are used for slightly different purposes in IKE and IPsec SA negotiations.



Note: StoneGate can also be set to comply with the FIPS 140-2 (Federal Information Processing Standard) requirements restricting the gateway capabilities to certain algorithms only. Using the FIPS mode means that MD5, Blowfish, Twofish, and CAST-128 are excluded from the list of supported algorithms.

Symmetric Encryption Parameters

Symmetric encryption is used to provide confidentiality of data during the SA negotiation based on IKE proposals. It is also used for encrypting/decrypting bulk data. StoneGate supports the following symmetric algorithms:

TABLE 10.1 Confidentiality parameters for VPN policy

Parameters	Description
DES (Data Encryption Standard)	Uses fixed 56-bit encryption keys and provides little security against brute force attacks
3DES (Triple DES)	Enhances security by applying DES three times to a single block of data using three different 56-bit keys
Twofish	Uses 128-bit encryption keys only
Blowfish	Uses 128-bit encryption keys by default, in fact key length can range from 40 to 448 bits
Rijndael (AES)	Uses 128-bit encryption keys by default
CAST-128	Uses 128-bit encryption keys only



Caution: DES is now considered insecure due to its short key length.

Data Integrity and Authentication Parameters

Authentication of the communicating parties and the integrity of the exchanged data is crucial for the reliability of the VPN implementation. StoneGate supports various methods for authentication and integrity validation.

Message Digest Algorithms

For data integrity, message digest algorithms, i.e., hash functions, are used. StoneGate features the following message digest algorithms:

TABLE 10.2 *Integrity parameters for VPN policy*

Parameters	Description
MD5 (Message Digest 5)	Widely used and fast, it produces a 128-bit message digest
SHA-1 (Secure Hash Algorithm 1)	Somewhat slower than MD5, it produces a 160-bit digest safer than MD5

Digital Signature Algorithms

Digital signatures based on public key encryption methods are used for verifying the identity of the security gateways. They are used also for validating certificates. For more information on StoneGate and digital certificates, see section “*StoneGate and Certificate Management*” on page 180.

The available algorithms for authentication are:

TABLE 10.3 *Authentication parameters for VPN policy*

Parameters	Description
RSA (Rivest Shamir Adleman) signature	The RSA algorithm uses a message digest to authenticate data with a digital signature.
RSA encryption	The RSA algorithm encrypts data used for authentication. Each party encrypts random data with the responder's public key. They can also decrypt data and calculate a message digest, which authenticates the IKE exchange.

TABLE 10.3 *Authentication parameters for VPN policy (Continued)*

Parameters	Description
DSS (Digital Signature Standard) signature	Uses the public key algorithm DSA (Digital Signature Algorithm) for digital signatures and SHA-1 for message digests. The key size varies from 512 to 1024 bits. The secret key operates on the message digest generated by SHA-1; the verification of the signature computes the digest again, applies the public key to decrypt the signature and then compares the results. The creation of DSS signature is roughly as quick as with RSA, but verification can be much slower.

Pre-shared secret

Additionally, an out-of-band pre-shared secret key can be used for authenticating IKE data traffic between security gateways. Both ends need to have the same key defined. The key exchange must be secured so as not to compromise the key to unauthorized parties.

Diffie-Hellman Parameters

The following two parameters can be set for computing Diffie-Hellman values:

TABLE 10.4 *Diffie-Hellman parameters for VPN policy*

Parameters	Description
Diffie-Hellman group for IKE	Specifying this group is required for computing the Diffie-Hellman value used for securing the IKE proposal and its keying material.
Diffie-Hellman group for PFS	Specifying this group is necessary for computing the Diffie-Hellman value used for securing the IPsec proposal and its keying material with PFS. The activation of PFS (see section <i>"Perfect Forward Secrecy"</i> below) includes in the new cryptographic material a newly computed DH value.

Perfect Forward Secrecy

Perfect Forward Secrecy (PFS) is an optional property of IKE transactions. It enhances the secrecy of keys and identities, but requires additional processing overhead. When PFS is set, a key used to secure the exchange of keying material cannot be used to compute

additional keys. PFS implies the calculation of the share secret key from scratch, so that there is no dependency between the old and the new key. Therefore, compromising a single key does not endanger the confidentiality of any other sessions of key exchange. With PFS, it is thus possible to ensure that a compromised key cannot be used to decrypt encrypted data that has been sent in the past.

In the *IKE negotiation mode*, administrators can optionally enable **PFS for identity**. The perfect secrecy of identities is achieved by deleting the resulting SA of the IKE SA negotiation. For every new IPsec SA needed, the firewall starts negotiating a new IKE SA. This is time-consuming and requires heavy computations.

In the *IPsec mode*, enabling the **Use PFS** setting triggers the computation of Diffie-Hellman values during the negotiation of IPsec SAs. The IKE SA being already negotiated, only the IPsec SA need to be renegotiated with an additional Diffie-Hellman exchange. The perfect secrecy of keys is achieved by deleting from the IKE SA the DH value derived from the IPsec SA proposal. Authenticated keying material from previous exchanges is therefore unrelated to keys negotiated for IPsec SA.

Lifetime Parameters

The lifetime of the IKE and IPsec tunnels can be specified in terms of elapsed time and transferred data. Lifetime (minutes or KB) represents the overall time (or data volume) after which the opened tunnels are closed. If a new tunnel is needed, the entire negotiation process is started all over again.

When an IPsec tunnel expires, only phase 2 negotiation is performed again based on the settings of the IPsec proposal and through the IKE negotiated tunnel. This process generates new key material to be used for the IPsec traffic. Similarly, IKE SAs are set to expire, but their lifetime is typically much longer than that of the IPsec SAs. The

IKE SA negotiation is also a more complex process which requires more time.

Path MTU Discovery

The Path Maximum Transmission (PMTU) discovery is used to adjust the packet size so that the packets can be routed without the need of fragmentation. When PMTU is enabled for a VPN, the overhead generated by the IPsec headers is taken into account and the maximum size for a plaintext packet that can fit into a IPsec packet is calculated. When a packet is send using VPN, its size is checked against the calculated maximum size. If it exceeds the maximum size, an ICMP message is sent to the originator of the plaintext packet advising to adjust the MTU value accordingly. The MTU of an external interface is assumed to be 1,500 bytes.

Hybrid Authentication

Hybrid authentication can be set for gateways that handle communications with mobile StoneGate VPN Clients. This allows the VPN Clients to authenticate themselves to the SGW with a username-password combination, while the SGW authenticates itself with a digital certificate. The other option is to use a purely certificate based authentication scheme. VPN Client authentication is discussed in *StoneGate Advanced Implementation and Beyond* course.

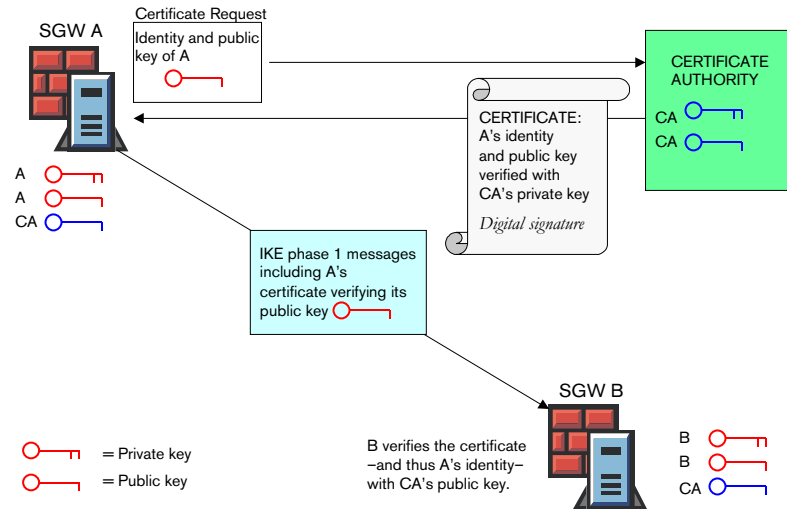
StoneGate and Certificate Management

With StoneGate, the security gateway authentication can be handled with the help of certificates signed and issued by CAs. The trusted CAs for each gateway can be defined in the VPN Manager, in the Security Gateway Properties window. Each gateway of a VPN is expected to present its own certificate signed by a mutually trusted CA, thereby proving that it owns the corresponding private key. Additionally, the communicating gateways need to have a copy of the CA's public key to be able to trust the CA and validate certificate

signatures issued from that CA. A certificate can be of several types (RSA with MD5, RSA with SHA-1, or DSA with SHA-1) and it is associated with a single security gateway.

The security gateways can trust only specified CAs. Each gateway requires a certificate with its public key, the trusted CA certificates, and the certificates of the remote nodes. The implementation of the security gateway (SGW) authentication based on certificates is illustrated in Figure 10.12.

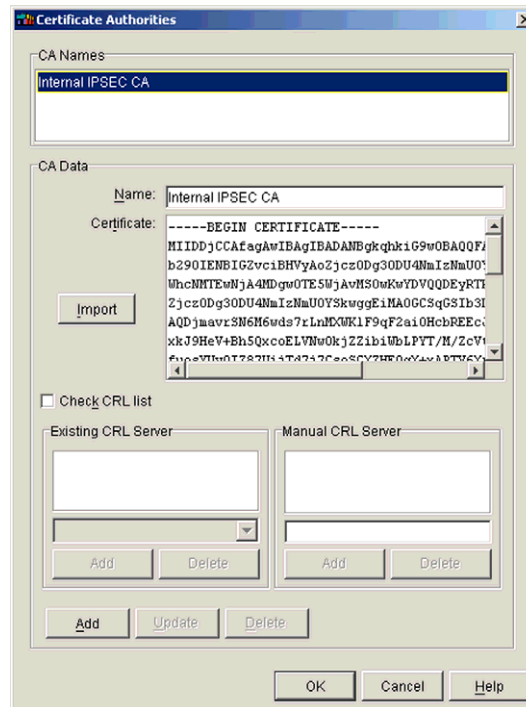
FIGURE 10.12 *Authentication with certificate*



The *certificate revocation lists* (CRL) managed by the CAs include the certificates that are reported as no longer valid after being lost, stolen or inapplicable. CRLs do not include expired certificates. Based on configuration properties, the security gateways can take care of accessing CRLs and checking the CRL validity by themselves. The full validation of a certificate normally requires to validate the CA

hierarchy from the local CA to the CA that issued the certificate. Facing the growing complexity of certification chain, CAs may support cross-certification.

ILLUSTRATION 10.5 *Certificate Authorities*



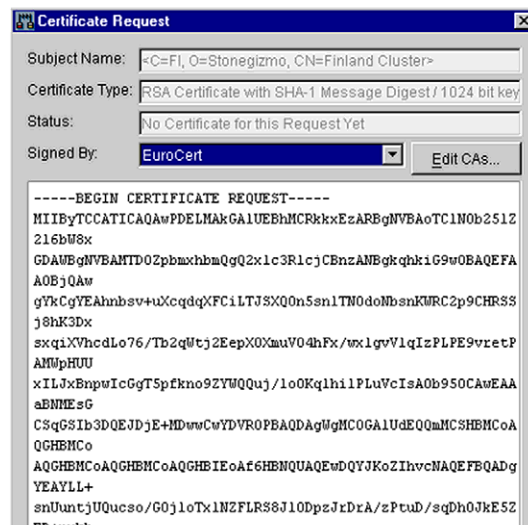
Certificate Requests

The certificate creation for the firewall cluster is initiated by the Management Server. The administrator provides the required information and creates a *certificate request* in the VPN Manager. Eventually, a firewall node generates a public-private key pair and synchronizes with the other nodes of the firewall cluster. The public key is added to the new certificate request. The generated certificate request file must then be send to a given CA by the administrator.



Alternatively, a request can be signed internally by the management system. After reception, the certificate signed with the CAs private key can be imported in the firewall cluster through the VPN Manager. The issued IPsec certificate is related to the IP addresses of the security gateway. The same certificate is available for all logical tunnels generated to and from that cluster, if they all use the same authentication method. Both ends of the VPN must have a certificate signed by a mutually trusted CA.

ILLUSTRATION 10.6 *Certificate Request*



Summary

In this chapter we have discussed the general principles of VPNs, and the related IPsec protocols. We have covered the basic procedures for encryption and authentication used to secure the VPN traffic. The Internet Key Exchange procedure, as well as certificate management principles were also introduced. All the required configurations for VPNs can be easily managed with StoneGate VPN Manager. Finally, we took a look at the different algorithms and parameters used by StoneGate for IPsec VPN traffic.

Review Questions

- What is the main difference between symmetric and asymmetric encryption?
- What different means are there to authenticate that the received data really is sent by the person who claims to have sent it?
- What is included in Security Associations?
- What kind of building blocks does a StoneGate VPN consist of?



LABS



StoneGate Installation

This lab exercise takes you step-by-step through the installation of StoneGate for a single firewall configuration. You should have a client machine and a firewall machine for this lab. Your instructor will provide you with an appropriate user ID and password for logging into the client machine for this installation.

Objectives

Upon completing this lab, you should have a single firewall installed and configured, as well as a Log Server, Management Server and GUI on a management system.

Getting Started

For this lab, you will need copies of both the StoneGate Management System software and the StoneGate Engine software. Your instructor will provide you with these on CD-ROM. First, you will install the management system and client on your 192.168.x.101 machine. When you have finished this, you will install the firewall engine on the other machine. Finally, you will configure the routing for StoneGate and define default routes. At the end of this lab, you will be ready to create and install a rule base.

Installing the Management System

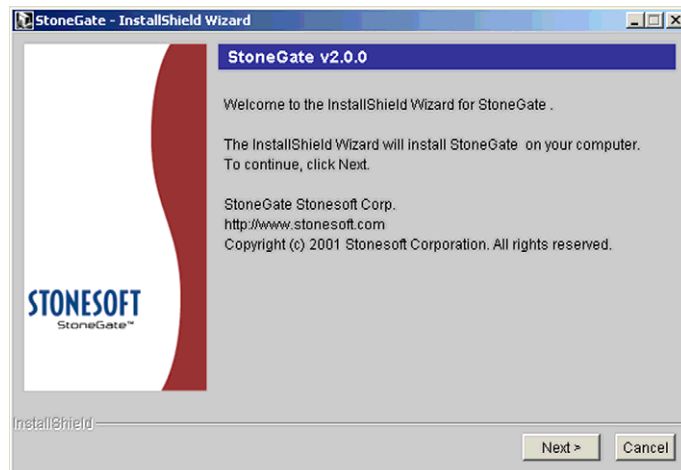
This section walks you through installation of the management system. To install the management system, perform the following actions:



1. Insert the StoneGate Management CD-ROM.
2. Locate and run `Setup.bat` for the StoneGate management system. For the class it should be located within the **StoneGate Management** shortcut folder on your desktop or from the CD-Rom Drive `x:\StoneGate_SW_installer\setup.bat`.

The InstallShield Wizard should begin to unpack the StoneGate installation files.

ILLUSTRATION 1.1 *Welcome screen*



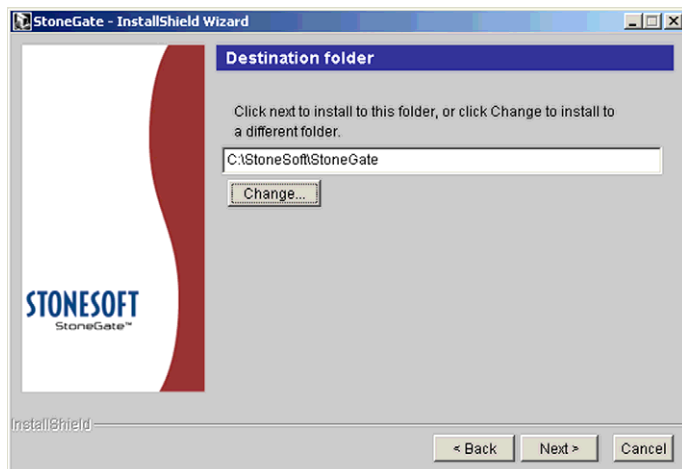
3. Once the Welcome screen appears, as shown in Illustration 1.1, click on **Next** to start the installation process.

ILLUSTRATION 1.2 *Installation license agreement*



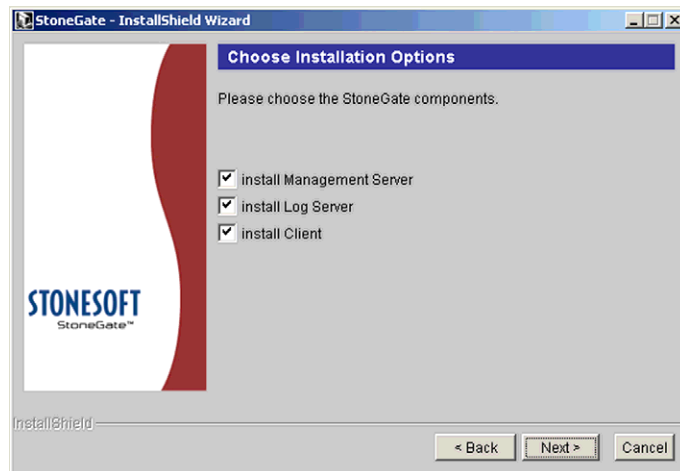
4. The License Agreement screen should appear next (Illustration 1.2). You must read and accept the license agreement, by clicking on the appropriate radio button, before you can continue. Once you agree to the license, click on **Next**.

ILLUSTRATION 1.3 *Destination folder location*



-
5. StoneGate will prompt you for a directory, into which it should install its files (Illustration 1.3). The default, C:\Stonesoft\StoneGate is acceptable, and should be used for class. Simply click on **Next** to continue.

ILLUSTRATION 1.4 *Components for installation*



6. After a confirmation dialog box, you are asked which options you want to install (Illustration 1.4). For this course, we will use all three options on a single machine, so accept the default by clicking on **Next**.

ILLUSTRATION 1.5 DBA account creation



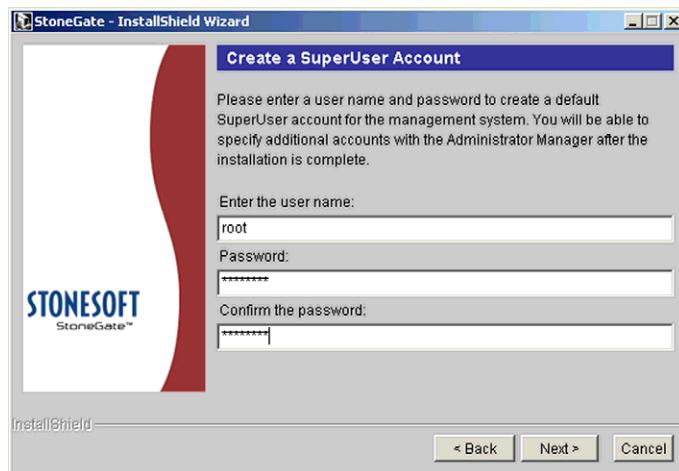
The screenshot shows a Windows-style dialog box titled "StoneGate - InstallShield Wizard". Inside, there's a section titled "Create a Database Account". The text inside says: "Please enter a user name and password to create an administrator account for the database engine. This account will be used by StoneGate to store and retrieve information in the embedded database engine." Below this, there are three input fields: "Enter the user name:" with the text "dba" entered, "Password:" with three asterisks "***" entered, and "Confirm the password:" with three asterisks "***" entered. On the left side of the dialog, there is a logo for "STONESOFT StoneGate™". At the bottom right, there are three buttons: "< Back", "Next >", and "Cancel".

7. StoneGate then allows you to specify the user name and password that will be used for the internal database engine (Illustration 1.5). Enter a name and then enter and confirm a password, then click on **Next** to continue.



Note: For this class, you should specify **dba** as the user name, and **dba** for the password as well.

ILLUSTRATION 1.6 Superuser account creation

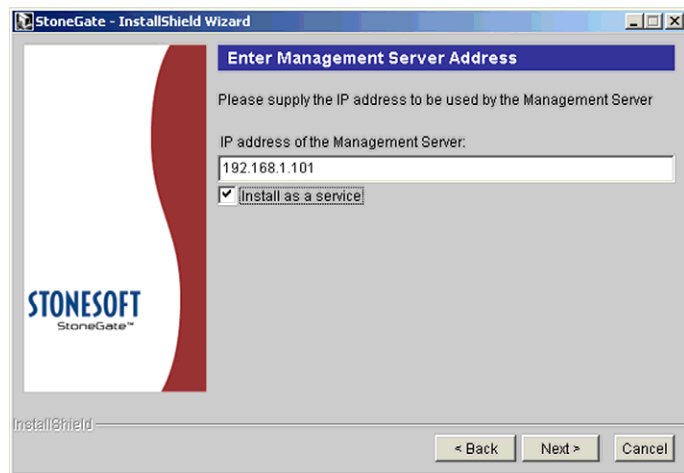


8. You can now define the default Superuser account (Illustration 1.6). Additional administrative accounts can be specified later, but StoneGate requires you to enter at least one account with Superuser privileges to begin. Enter the account information, then click on **Next**.



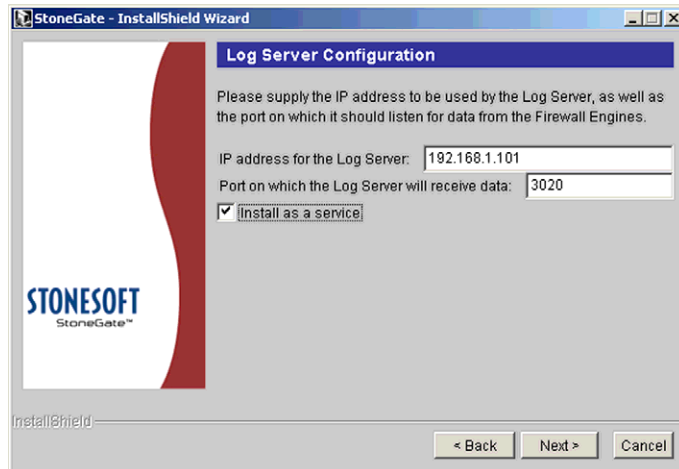
Note: For this course, use **root** for the initial Superuser, and **password** as the password for this account.

ILLUSTRATION 1.7 *Management Server address*



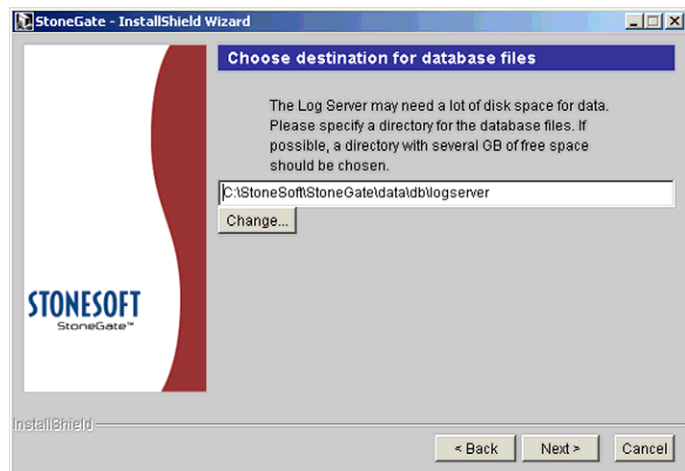
9. StoneGate then asks which IP address should be used to configure the Management Server (Illustration 1.7). A list of existing IP addresses discovered on the machine are presented, or you can specify an IP address. You also have the option to install the server as a service by selecting the check box. Select the appropriate Management Server address for your site (**192.168.x.101**), ensure that the **Install as a service** check box is selected, then click **Next**.

ILLUSTRATION 1.8 *Log server configuration*



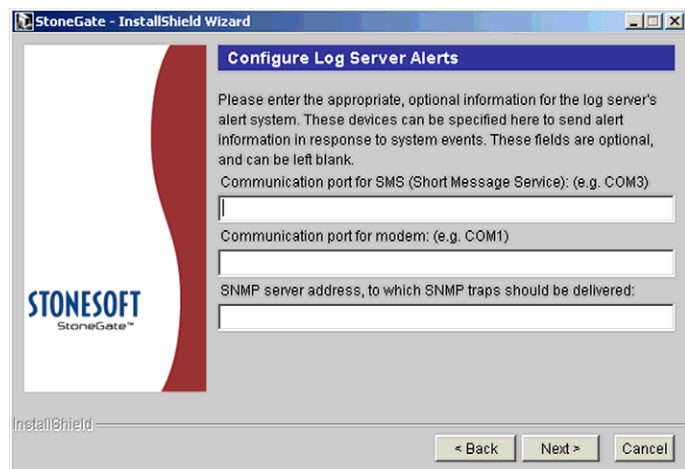
10. The next component to configure is the Log Server (Illustration 1.8). The installation will present a list of existing IP addresses discovered on the machine, or you can specify an IP address. You may also specify a port it should use to listen for log data from the firewall engines. In this class, the Log Server is also on the management system, so select the same IP address again (**192.168.x.101**), mark the **Install as a service** check box, and leave the default port of 3020, then click on **Next**.

ILLUSTRATION 1.9 *Log server database path*



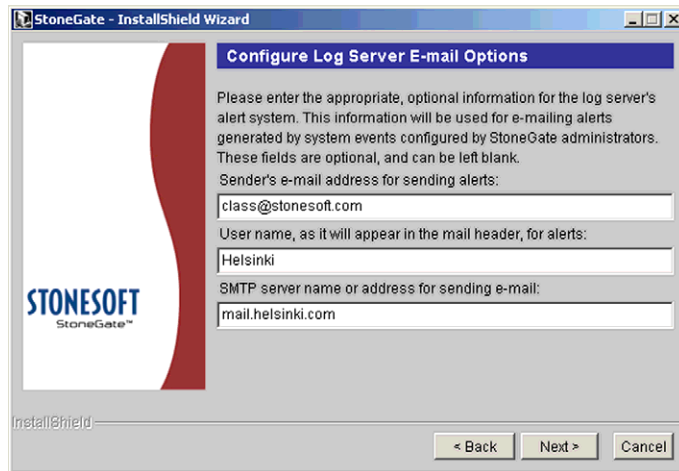
11. Because logging can be quite intensive, StoneGate will provide you with an option to locate the Log Server's database on a separate partition or disk from the Management Server's database. Specify the location for the Log Server's database files next, as shown in Illustration 1.9, then click **Next**.

ILLUSTRATION 1.10 *Log server alerts*



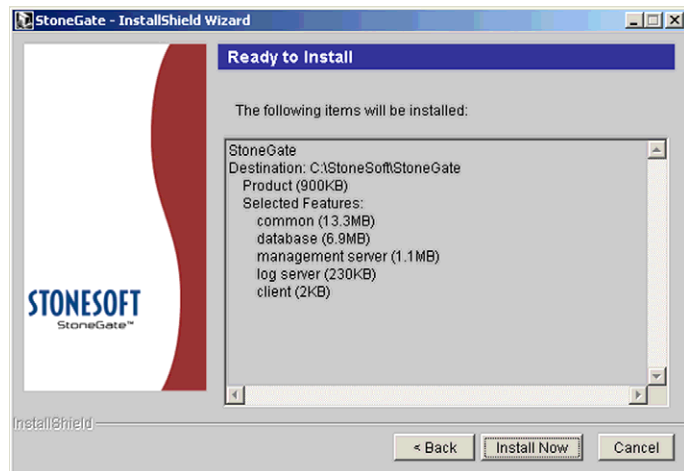
-
12. StoneGate will now ask you for additional information to configure the defaults to use for services when sending alert information. For this first screen, ensure all three fields are left blank, then click on **Next**.

ILLUSTRATION 1.11 *E-mail options*



13. You can now define the Log Server e-mail options using the following data, then click on **Next**:
- Sender e-mail address for sending alerts: **class@stonesoft.com**
- User Name: **<site name>**, (*e.g., Helsinki*)
- SMTP server: **mail.<site>.com** (*e.g., mail.helsinki.com*)

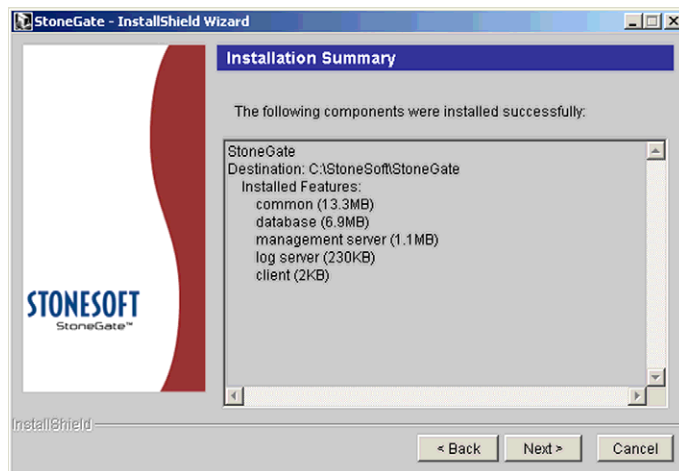
ILLUSTRATION 1.12 *Installation confirmation*



14. StoneGate will respond by confirming you are ready to install. The window will include a list of items to be installed, and will direct you to select to return **Back** to change information, **Cancel** to abort this process, or **Install Now**. Click on **Install Now**.

StoneGate will now install the packages and configure the product as you specified.

ILLUSTRATION 1.13 *Installation complete*



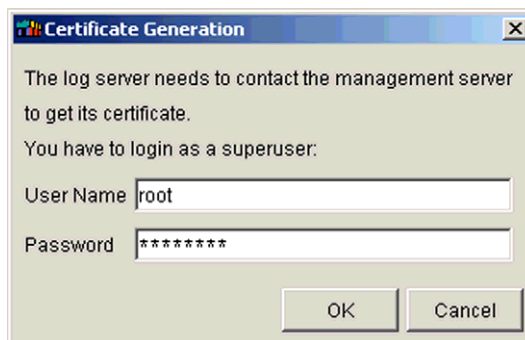
15. Once the Installation is complete, click on **Next** to exit.

Certify the Log Server

Because the Log Server can be installed on a separate machine than the Management Server, the two systems communicate via SSL and trust each other through keys and certificates. In order to run, the Log Server must obtain a key and certificate from the Management Server. This section will outline the steps required to get the certificate for the Log Server.

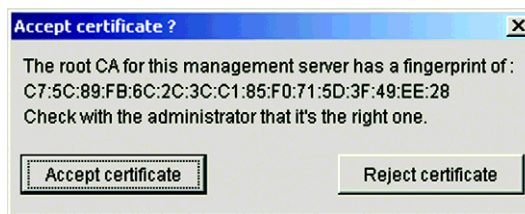
1. Before you can request a certificate, you must ensure that the Management Server is up and running. To start the Management Server:
 - Run **Start → Programs → StoneGate → Windows Service Panel**.
 - Once the Services control panel is open, locate the **StoneGate Management Server** service. Click on it, then click on the **Start Service** button.
2. Run **Start → Programs → StoneGate → Request Log Server Certificate**.

ILLUSTRATION 1.14 *Log certificate request - Superuser login*



3. The certificate request asks for a Superuser account to login as (Illustration 1.14); use the **root** account to login to the management system.

ILLUSTRATION 1.15 *Certificate acceptance*



4. The certification request will then login to the management system with the credentials you specified. It will obtain a certificate, and present some information for your approval, as seen in Illustration 1.15. On the resulting screen, select **Accept certificate**.

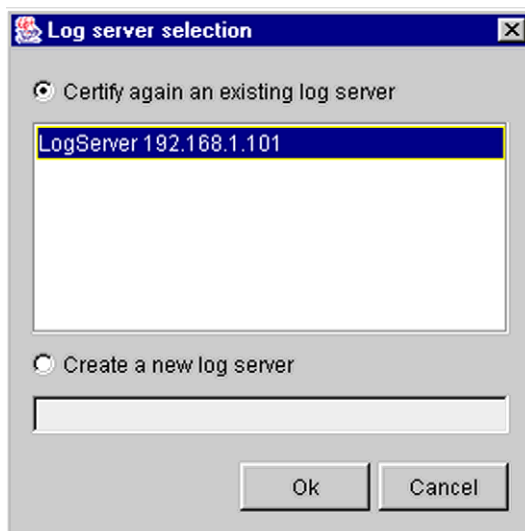


.....

Tip: You can verify the fingerprint of the certificate by running the *Show Fingerprint* command on the Management Server. Go to **Start → Programs → StoneGate → Show Fingerprint**.

.....

ILLUSTRATION 1.16 Log Server Selection



5. The Log server selection window will appear on screen. Mark the check box **Certify again an existing Log Server** and click on **LogServer 192.168.x.101** as seen in Illustration 1.16. On the resulting screen, click **Ok**.

ILLUSTRATION 1.17 Certificate installed



6. You should then receive a notification that the certificate of the chosen server has been regenerated and installed successfully (Illustration 1.17).

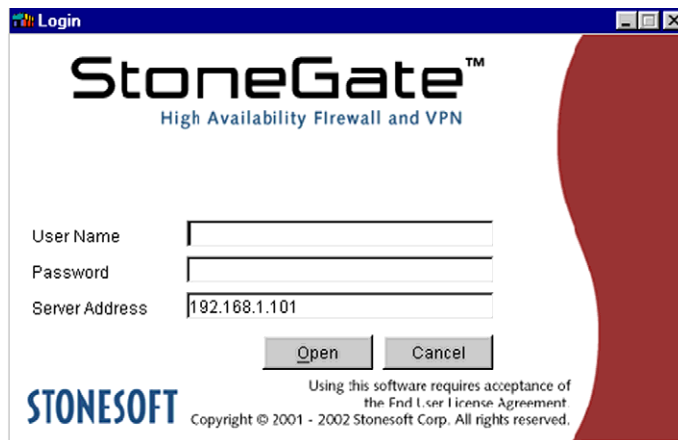
The Log Server is now prepared to communicate with the Management Server and any assigned firewalls.

Install Licenses

Now that the management system is installed and the Log Server is certified, the next step is to login to the management system and install your license file. Your instructor will provide you with the license files, or information on where they are located on your computer.

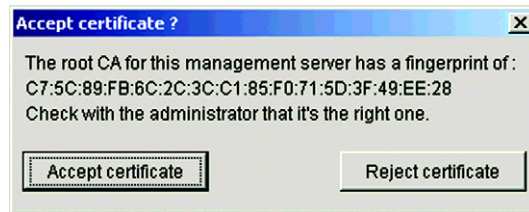
1. To connect to the Management Server, you should run the GUI and login. To start, go to **Start** → **Programs** → **StoneGate** → **Administration Client**.

ILLUSTRATION 1.18 *Management system login*



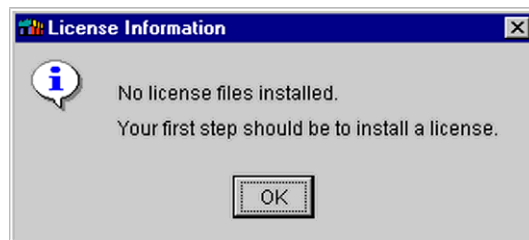
-
2. Login to the StoneGate management system, using the Superuser account you created in section “*Installing the Management System*”, as shown in Illustration 1.18.

ILLUSTRATION 1.19 *Certificate acceptance*

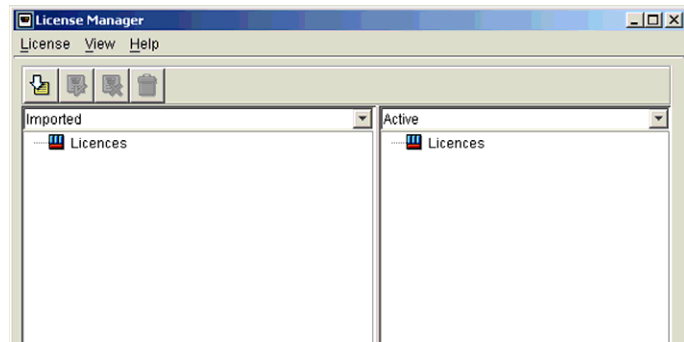


3. Because this is the first time the GUI has contacted the Management Server, they perform a certificate and key exchange, much like the Log Server certificate request you just performed in section “*Certify the Log Server*” on page 198.

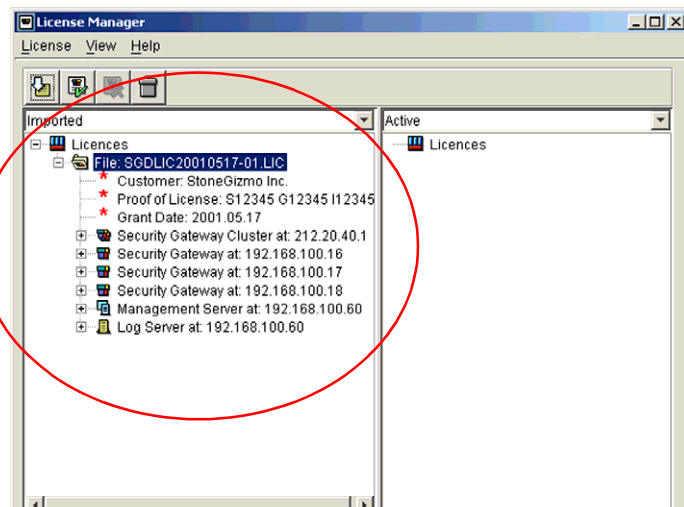
ILLUSTRATION 1.20 *License Information Window*



4. Before login in, the License Information window appears on screen to inform you that no license files are installed. Click **OK**.
5. Once you’ve logged in, StoneGate will start the License Manager automatically, since you must enter a license before you can perform any other actions. At this point, the License Manager is the only icon that is active on the Launchpad, and you can start it from there if it doesn’t start automatically. See Illustration 1.21.7

ILLUSTRATION 1.21 *StoneGate License Manager*

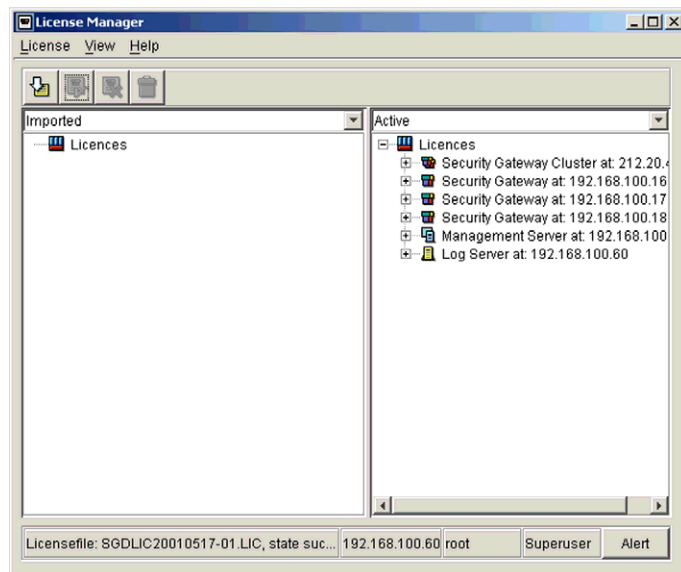
6. In the License Manager, click on the **Import Licenses** button on the toolbar, or choose **License** → **Import**.
7. A file browser should appear, allowing you to locate and select the StoneGate license file for your machine. Choose the appropriate *sitename*.JAR file, and click **Open**.

ILLUSTRATION 1.22 *An imported license in the License Manager*



8. The license file you opened should be displayed as a tree of information on the left panel of the License Manager, under Imported licenses. Illustration 1.22 shows a sample license. Verify the information for the license you have imported, and then click on the **Activate** button between the two main panels.

ILLUSTRATION 1.23 *An active StoneGate license*



9. Once you click on the **Activate** button, the license information should migrate to the right panel, and be displayed under Active licenses. You have successfully installed your StoneGate license!
10. Close the License Manager. At this point, the remaining managers on the Launchpad become activated.

Defining a StoneGate Firewall

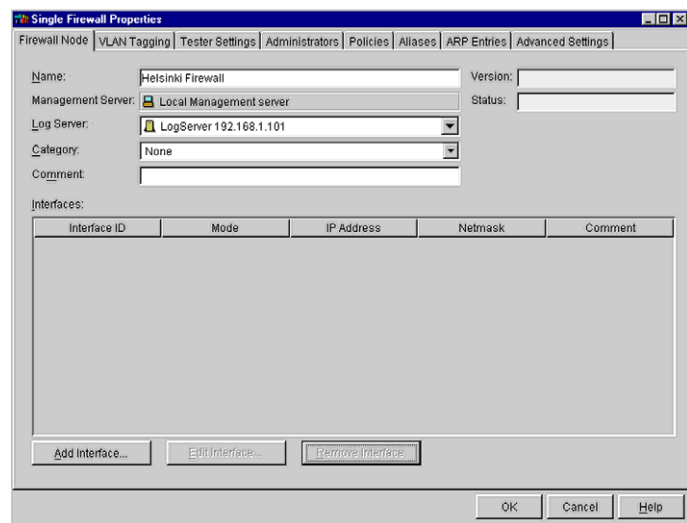
Create a Single Firewall Element

Now that you have licensed your system, you can proceed to define your firewall. StoneGate enables the administrator to perform all configuration tasks through the user interface, instead of having to configure each firewall engine individually. To define your firewall perform the following actions.



1. Open the Network Element Manager by clicking on its icon from the StoneGate Launchpad.

ILLUSTRATION 1.24 *Single Firewall Node dialog box*

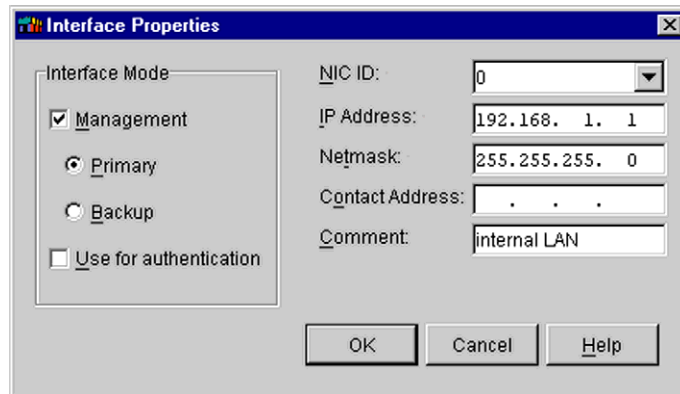


2. Choose to create a new single firewall, either by right-clicking on the Firewalls listed in the Repository View of the left panel, or by clicking on the Single Firewall icon on the element toolbar. The Single Firewall Node Properties dialog box opens, as seen in Illustration 1.24.

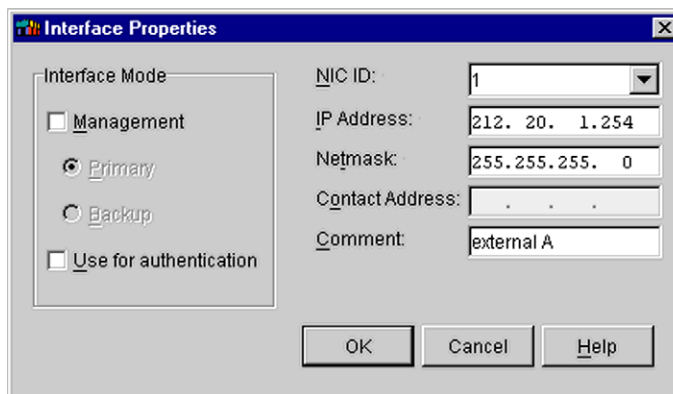


3. In the Single Firewall Node dialog box, enter a name for your firewall, select the Log Server from the selection box, and enter a comment if you desire.

ILLUSTRATION 1.25 *Properties for network interface 0*



4. Click on the **Add Interface** button to begin defining your network interfaces. For NIC ID 0, enter the internal address for your firewall (**192.168.x.1**). It should look similar to the sample depicted in Illustration 1.25. Be sure to also check the box that this interface is a Management interface (primary). Click on **OK**.

ILLUSTRATION 1.26 *Properties for network interface 1*

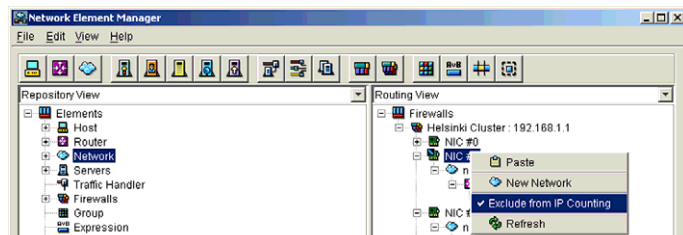
5. Add an additional interface, this time for NIC ID 1, and specifying the IP address information for the external ISP (**212.20.x.254**). Illustration 1.26 shows an example. Click on **OK**.
6. After you have created both interfaces, click on **OK** in the Single Firewall Node dialog box. The firewall you just defined should show up under Firewalls in the Repository View of the Network Element Manager.

Exclude from IP Counting

If your license is based on IP Counting, you must exclude one interface from IP counting, or you will be unable to upload a rule base to your firewall engines. To exclude an interface from IP counting, do the following:

1. Ensure that you are in the Routing View of the Network Element Manager with the view expanded.
2. Right-click on your external NIC, and select **Exclude from IP Counting** in the contextual menu.

ILLUSTRATION 1.27 IP Counting



Save Initial Configuration

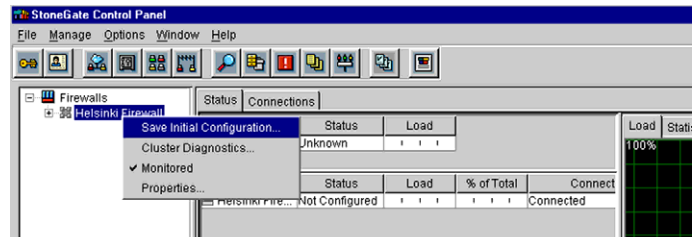
Now that you've defined the properties of your firewall element, you can save the configuration information so that you can begin installing the actual firewall engine. In this task you will instruct StoneGate to create the required configuration files that the engine will need during the next phase of installation. StoneGate will also create a one-time password that can be used by the engine to obtain its configuration. To save the initial configuration, perform the following tasks.

1. In the StoneGate Control Panel, right-click on the firewall you just defined. In the contextual menu that appears, choose **Save Initial Configuration**, as shown in Illustration 1.28.



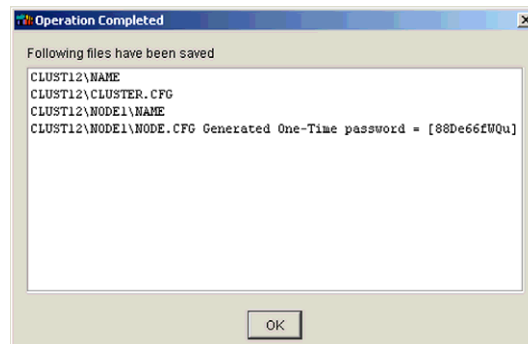
Tip: You need to ensure you are in the StoneGate Control Panel and not the Network Element Manager in order to save the initial configuration.

ILLUSTRATION 1.28 Saving the initial configuration



2. StoneGate will present you with a dialog box to select a directory where it will save the files. For this class, insert the provided floppy disk in drive A: and browse to the directory **A:**.

ILLUSTRATION 1.29 Saved files dialog with one-time password



3. Once StoneGate has completed the save, it will display an Operation Completed dialog, as seen in Illustration 1.29. In this box, it will list the files it created, as well as the one-time password. Make a note of the one-time password, as you will need it during the engine installation. You can leave this dialog box open while you proceed with the engine install in section “*Installing the Firewall Engine*” below.



.....
Caution: Do not close this dialog before making a note of the one-time password, or completing the engine install!
.....

Ensure that the Log Server is Started

Before moving on to the next step, installing your firewall engine, you should first ensure that your Log Server is started and running when you need it later. To do this:

1. Double-click the Services icon on your desktop. If it is not on your desktop, you can find it in your Windows Control Panel.
2. In the services box, locate the **StoneGate Log Server** and select **Start**.
3. Click **Close**.

Installing the Firewall Engine

Once you have the one-time password, and the engine is defined in the management system, you are ready to install the engine. StoneGate firewalls have the operating system and the firewall built in, and you will install both in this process. The operating system for StoneGate is based on Linux, so some familiarity with Unix systems is helpful, although not required.

Basic Configuration



Insert CD

1. To begin, insert the StoneGate engine installation CD-ROM into the drive and reboot the machine.

ILLUSTRATION 1.30 *Installation Options*

```
StoneGate Engine Installation System
Existing StoneGate installation has not been detected.
1. Full install
2. Full install in expert mode
Enter your choice: 1_
```

2. Once the system boots from the CD-ROM, you will be prompted to choose between two types of installations: **Full Install** and **Full Install in expert mode**. For class, type in 1, **Full Install** option as seen in Illustration 1.30.

ILLUSTRATION 1.31 *CPU*

```
StoneGate Engine Installation System
Existing StoneGate installation has not been detected.
1. Full install
2. Full install in expert mode
Enter your choice: 1
Select the number of CPUs this machine has:
1. One (uniprocessor)
2. Two or more (symmetric multiprocessing)
Enter your choice: 1_
```

3. After selecting option 1, you are asked to select the number of CPUs of your machine. Type in 1, **One (uniprocessor)**. See Illustration 1.31.

ILLUSTRATION 1.32 *Continue*

```
StoneGate Engine Installation System

Existing StoneGate installation has not been detected.

1. Full install
2. Full install in expert mode

Enter your choice: 1

Select the number of CPUs this machine has:

1. One (uniprocessor)
2. Two or more (symmetric multiprocessing)

Enter your choice: 1

Hard disk will be partitioned automatically and all existing data will be lost.
Type YES to continue and NO to cancel.

Do you want to continue? YES_
```

4. Next, you are informed that your hard drive is about to be configured automatically. Type **YES** to continue.

ILLUSTRATION 1.33 *Installation Finished*

```
StoneGate Engine Installation System

Existing StoneGate installation has not been detected.

1. Full install
2. Full install in expert mode

Enter your choice: 1

Select the number of CPUs this machine has:

1. One (uniprocessor)
2. Two or more (symmetric multiprocessing)

Enter your choice: 1

Hard disk will be partitioned automatically and all existing data will be lost.
Type YES to continue and NO to cancel.

Do you want to continue? YES
Partitioning hard disk...
Creating filesystems...
Extracting StoneGate image...
Installing boot loader...
Installation finished! Remove CDROM disc and press Enter to reboot._
```

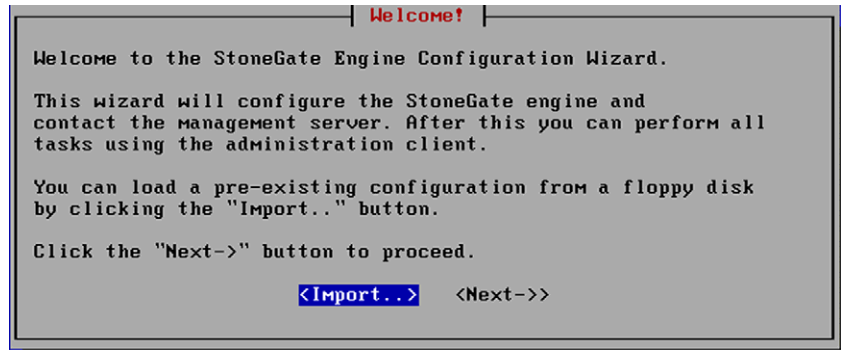


Remove CD

5. StoneGate begins the automatic installation process. When finished, you will be prompted to remove the CD-ROM installation disc and press ENTER to reboot.

Select Method of Configuration

ILLUSTRATION 1.34 *StoneGate Engine Configuration Wizard*



1. After removing the CD-ROM and rebooting, you will continue the installation process by configuring your firewall engine. The StoneGate Engine Configuration Wizard opens. At this point, you have the option to import the engine's initial configuration by floppy or do it manually.
 - 1.1 Earlier, you created a floppy disk when you saved the initial configuration of your firewall. You can now insert this disk in the floppy drive. Select **Import** and press ENTER to continue.

ILLUSTRATION 1.35 *Insert Floppy disk*



2. The Info window appears on screen. Insert the floppy disk and press **Ok**.



Note: If you have saved more than one configuration in the floppy disk, a selection window will appear on screen. Select the appropriate initial configuration for the node you are installing and press ENTER.

Configure OS Settings

ILLUSTRATION 1.36 Configure OS Settings

```
Step 1 of 3: Configure OS settings

Keyboard layout: <US English>
Local timezone: <unset>

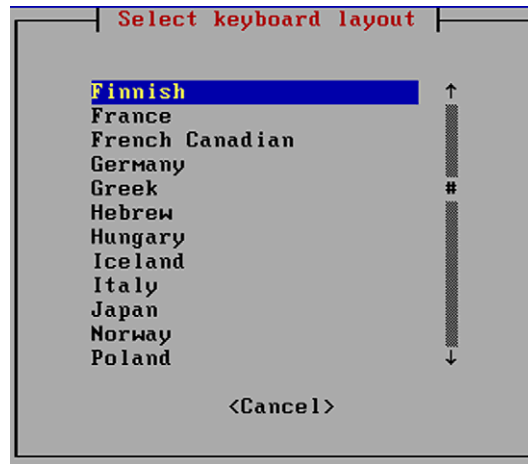
Host name: Helsinki node

Root password:
Enter:
Re-enter:

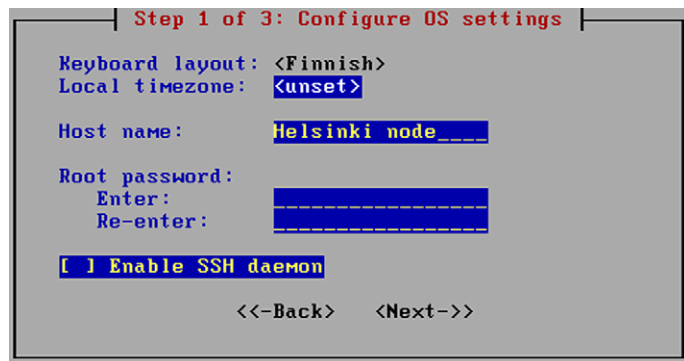
[ ] Enable SSH daemon

<-Back    Next->
```

1. The first Configure OS settings window opens with the host name of the node you imported. From here you can configure such things as keyboard layout, host name, password and whether to enable the SSH daemon. To begin, highlight the Keyboard layout line and press ENTER. See Illustration 1.36.

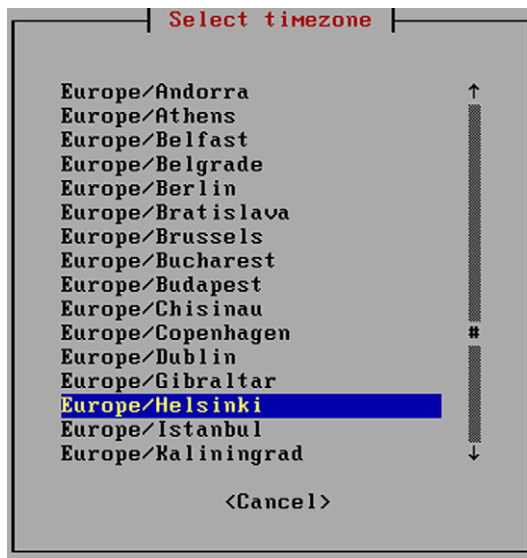
ILLUSTRATION 1.37 *Select Keyboard Layout*

2. The Select Keyboard window will be opened. Highlight the appropriate keyboard layout and press ENTER to continue.

ILLUSTRATION 1.38 *Local Timezone*

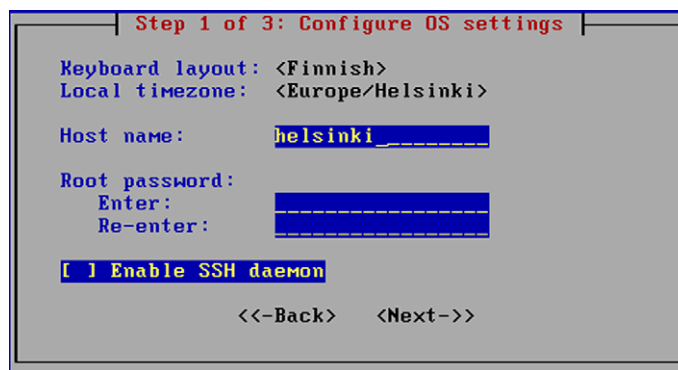
3. You will return to the Configure OS settings window. Next, highlight the Local timezone line and press ENTER.

ILLUSTRATION 1.39 *Select Timezone*



4. Select your timezone and press ENTER..

ILLUSTRATION 1.40 *Host name*



5. Once again you return to the Configure OS settings window. As you can see, the host name has been imported from the floppy disk. In this example, the name is **helsinki node**, the name of your

firewall will vary depending on your classroom situation. For the class, the host name will be just the name of the city. See Illustration 1.40.



.....
Caution: The Host name is only allowed to contain letters, numbers, and underscores.
.....

ILLUSTRATION 1.41 Root password

```
Step 1 of 3: Configure OS settings

Keyboard layout: <Finnish>
Local timezone:  <Europe/Helsinki>

Host name:       helsinki

Root password:
Enter:           *****
Re-enter:        *****

[ ] Enable SSH daemon

<-Back    Next->
```

- Next, StoneGate asks you to assign a root password, which you can use to login to the engine's console. For the class, use **password** as the password for the root account. And then, re-enter the password on the line below. See Illustration 1.41.

ILLUSTRATION 1.42 *Enable SSH daemon*

Step 1 of 3: Configure OS settings

Keyboard layout: <Finnish>
Local timezone: <Europe/Helsinki>

Host name: helsinki

Root password:
Enter: *****
Re-enter: *****

[*] Enable SSH daemon

<<-Back <Next-->

7. You will then need to decide whether to enable the SSH daemon and allow SSH contact to the engine. By default this feature is not enabled. For class, enable the daemon by highlighting the line and pressing the space bar. An asterisk (*) will appear between the brackets to show that the daemon has been enabled as seen in Illustration 1.42.
8. Highlight **Next** and press ENTER.

Configure Network Interfaces

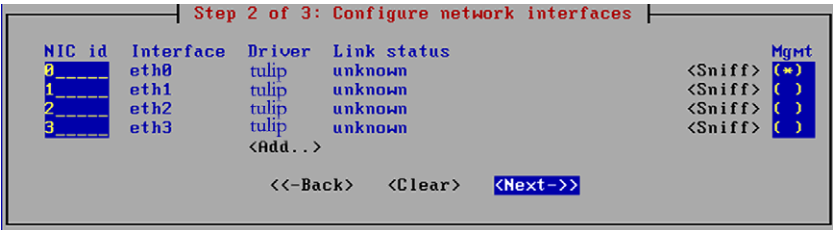
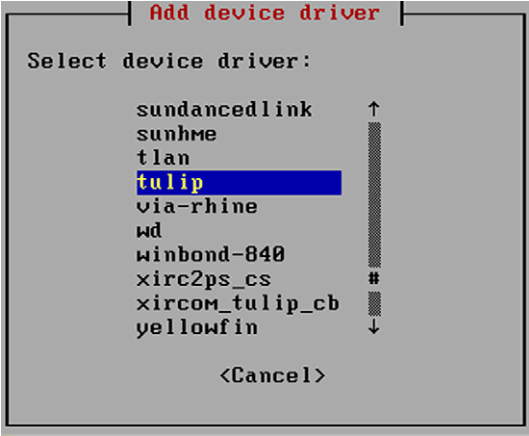
ILLUSTRATION 1.43 *Configure Network Interfaces*

Step 2 of 3: Configure network interfaces

NIC id	Interface	Driver	Link status	Mgmt
		<Add...>		

<<-Back <Clear <Next-->

1. Next, you will see the network interfaces window. Highlight **Add** and press ENTER in order to select a device driver.





Tip: The Sniff options are available in case you have trouble connecting to the Management Server after configuration is complete. They allow you to troubleshoot and detect whether you have assigned your NIC ids correctly.

Management Server Contact

ILLUSTRATION 1.46 Management Server Contact Information

Step 3 of 3: Prepare for management contact

Enter data for switching to the initial configuration and/or contacting the management server. Fields marked with * must be filled.

☒ Switch to initial configuration

Firewall node

IP address:* 192.168.1.1

Netmask:* 255.255.255.0

Gateway to management: _____

☒ Use VLAN

VLAN identifier: _____

☐ Contact management server

Management server

IP address:* 192.168.1.101

One-time password:* 525d86Nabk_

Key fingerprint: 9F:F2:73:74:9C:B7:BB:E7:9B:6D:34:63:27:5F:34:3A_

<<-Back> <Finish>

- Next, the Prepare for management contact window opens. When you first install the engine, **Switch to initial configuration** is checked by default because there is no active configuration. Because this is the initial configuration, leave the box checked.



Note: If you run the `sg-reconfigure` command later, you can select whether to switch to a new initial configuration by selecting the box, or you can choose to use the current configuration by unchecking the box. If you leave the box unchecked, the currently active security policy will remain active. All other changes (host name, time zone, SSH daemon, NIC mapping, management contact, etc.) will take effect after selecting **Finish**.

5. Next, verify that all the information of the firewall node and Management Server is correct.
 - 5.1 Because your management and firewall are on a directly connected network, you do not need to insert a gateway to the management. Leave this blank.
 - 5.2 For this class we are not using VLAN tagging. Leave the VLAN identifier field blank.

ILLUSTRATION 1.47 Contact Management Server

```

Step 3 of 3: Prepare for management contact

Enter data for switching to the initial configuration and/or contacting
the management server. Fields marked with * must be filled.

[*] Switch to initial configuration
Firewall node
  IP address:*          192.168.1.1
  Netmask:*            255.255.255.0
  Gateway to management:
  [ ] Use VLAN
    VLAN identifier:
  [-] Contact management server
Management server
  IP address:*          192.168.1.101
  One-time password:*   525d86NAbk
  Key fingerprint:     9F:F2:73:74:9C:B7:BB:E7:9B:6D:34:63:27:5F:34:3A
                                <<-Back    <Finish>
  
```

6. Select **Contact Management Server** by highlighting the brackets and pressing SPACE BAR. Ensure that an asterisk (*) fills the space as

seen in Illustration 1.46. When you have finished verifying all the relevant contact information highlight **Finish** and press ENTER.

ILLUSTRATION 1.48 *Installation Successful*

```
h2a_liblogger_init: c699be00, logdev0: c699be00
h2a_logdev_init:msgbuf_init
h2a_logdev_init:waitqueue
h2a_logdev_init:ok
h2a_logdev_init:msgbuf_init
h2a_logdev_init:waitqueue
h2a_logdev_init:ok
SSH IPSEC version 4.1.1 built on May 14 2002 07:44:27
init_module: registering hook nro 0.return=0
init_module: registering hook nro 1.return=0
init_module: registering hook nro 2.return=0
init_module: registering hook nro 3.return=0
init_module: done
SG: cluster not configured
Contacting management system...
Contact succeeded.
root@Helsinki:~# _
```

7. After this, the node tries to connect to the Management Server. After a brief period, the Management Server should show the connection as “Connected” in the StoneGate Control Panel **Monitoring Status** tab.

Summary

At this point you should have successfully completed the following tasks:

- installed the management system and GUI
- certified the Log Server
- installed a license
- defined a single firewall engine
- installed a firewall engine

Once these steps have been completed, you are ready to move on to the next unit, *Chapter 4* on page 67.

Configuring Routing and Anti-Spoofing

This lab takes you step by step through configuring routing and anti-spoofing for your StoneGate firewall. At this point you should have a management system and a single firewall engine installed. The firewall engine should have successfully contacted the Management Server, yet no policy has been installed.

Objectives

After completing this lab, you should have correctly:

- defined a router element
- created a default route
- checked and modified the anti-spoofing measures for your firewall.

Getting Started

In order to create routes and check anti-spoofing settings, you need to use the GUI. StoneGate allows easy control and access of all firewall functions, including those tasks where traditional firewalls required you to work on the engines directly.

To get started, you should ensure that the Management Server and Log Server are running, and then run the StoneGate Administration

Client, if it is not already started. Once you have logged into the management system, you can begin to perform the following tasks.

Defining the Routing

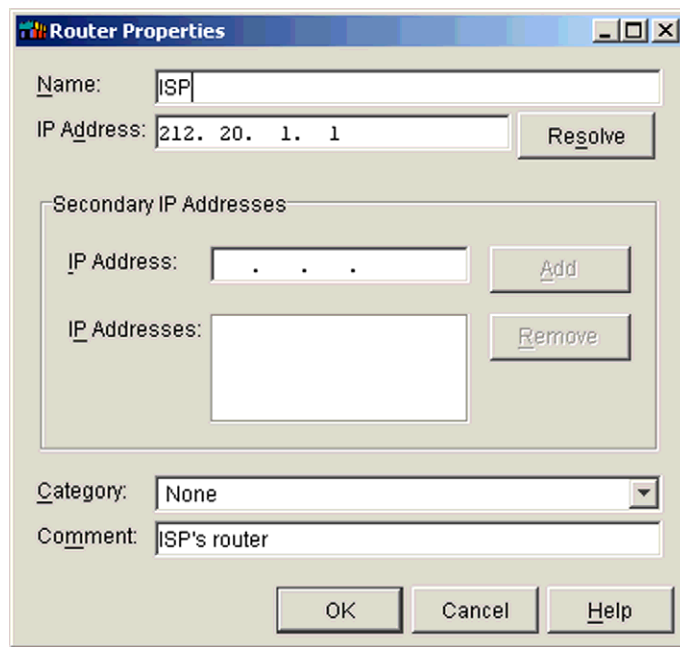
Define a Router

To define routes, you need to create at least one router element. This section will outline how to create a router that can be used for defining routes.



1. Start the Network Element Manager from the StoneGate Control Panel, if it is not already running.
2. Click on the Router icon on the toolbar, right-click on the Router in the Repository View panel, or use **Edit** → **New** → **Router** from the menu bar.

ILLUSTRATION 2.1 *Router Properties window*

A screenshot of the 'Router Properties' dialog box. The window has a title bar with the text 'Router Properties' and standard window controls. Inside, there are several fields: 'Name:' with the value 'ISP', 'IP Address:' with the value '212. 20. 1. 1' and a 'Resolve' button to its right. Below these is a section titled 'Secondary IP Addresses' containing an 'IP Address:' field with three dots, an 'Add' button, and an 'IP Addresses:' list box with a 'Remove' button. At the bottom, there is a 'Category:' dropdown menu set to 'None' and a 'Comment:' text field with the value 'ISP's router'. At the very bottom are 'OK', 'Cancel', and 'Help' buttons.

Router Properties

Name:

IP Address:

Secondary IP Addresses

IP Address:

IP Addresses:

Category:

Comment:

3. Once the Router Properties window appears, you can define your router element by filling in the boxes. Enter a name and IP address for the router. For class, enter the IP address of your ISP (212.20.x.1). Then click on **OK**.

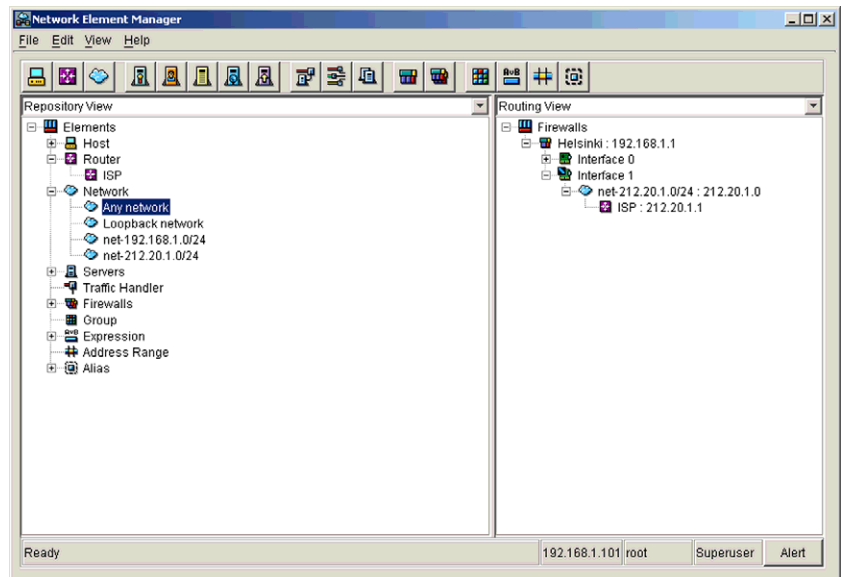
You can now use this router to create routes in the Network Element Manager.

Create a Default Route

To create a default route, you use the newly created Router element.

1. In the Network Element Manager, change the view panel on the right side to the Routing View.
2. Expand the tree in the Routing View by clicking on the plus signs (+) next to each item, until the network is displayed below each interface card.

ILLUSTRATION 2.2 *StoneGate's Routing View panel*



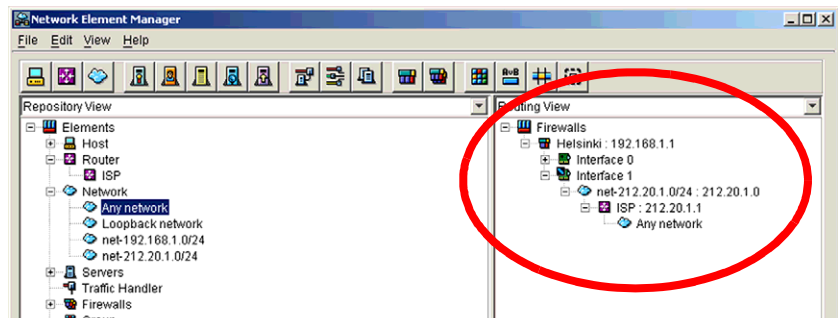
3. Take your new Router element from the Repository View on the left panel and drag it on your external network element in the Routing View on the right panel. The Network Element Manager should now resemble Illustration 2.2.



**Invalid
Router**

Note: If you drag the router in a location that is not logically correct — for example, the internal network — StoneGate will signify your error by placing an exclamation point over the router element in the Routing View.

ILLUSTRATION 2.3 Default route in the Routing View



4. Now you need to tell StoneGate what networks are available to StoneGate through that router. Since we want to create a default route, we will use the **Any Network** element. Take the Any Network element from the Repository View on the left panel, and drag it on the Router element in the Routing View on the right panel.



Tip: If you inadvertently place an element in a tree at the wrong location, you can remove it again by right-clicking on the element and selecting **Remove**.

You've now defined the default route for StoneGate. Additional routes, both static and default, can be added in the same manner.

Anti-Spoofing View

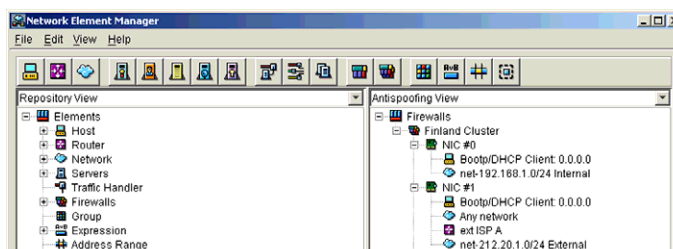
Modify Anti-Spoofing

Traditional firewall products required an administrator to configure anti-spoofing rules on interfaces by hand, often leading to errors or oversights. With StoneGate, anti-spoofing is built into the product, and enabled by default.

Although you no longer have to enable or configure basic anti-spoofing measures by default, you may find a need to customize the default anti-spoofing definitions. To do so, you can easily use the Network Element Manager and its Anti-Spoofing View.

1. In the Network Element Manager, change the view on the right panel to the Anti-spoofing View.

ILLUSTRATION 2.4 The Anti-spoofing View in the Network Element Manager



2. Expand the tree in the Anti-spoofing View by clicking on the plus signs (+) to the left of each element. The view should resemble the one depicted in Illustration 2.4.
3. Create a new Host element, by clicking the Host icon on the toolbar. In the Host Properties window, define the IP address of the Internet Web server at **192.89.38.200**, and name it **Instructor**.
4. Take the new host Instructor from the Repository View in the left panel, and drag and drop it to the Anti-spoofing View under **NIC #1** in the right panel.

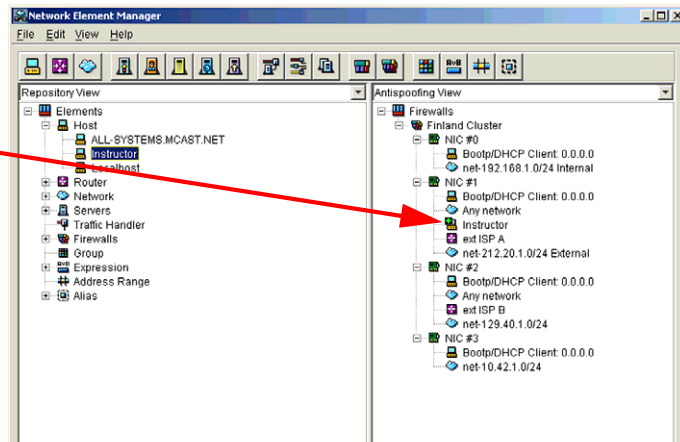


Note: Elements added manually to the Anti-spoofing View will be displayed with a plus sign (+) to distinguish them from the automatically generated anti-spoofing elements.



ILLUSTRATION 2.5 *An extended anti-spoofing definition*

**Host
added**



Your Anti-spoofing View should now resemble the one depicted in Illustration 2.5. You can now close the Network Element Manager and proceed with the next unit, *Chapter 5* on page 81.

Summary

During this lab you have learned how to configure StoneGate routing. This is easily done by creating routing elements, and then simply dropping them onto their respective network elements (which are created automatically) in the Routing View. You can create a default route by dropping the Any Network element onto your router element. Finally, you learned how to modify your anti-spoofing by creating host elements and adding them to the appropriate NIC in the Anti-spoofing View.

Once you have completed this lab, you are ready to create basic policies and install them on your firewall in Lab 3.

STOP. Wait for your instructor before continuing.

Creating Basic Policies

This lab takes you through the creation of basic security rules in StoneGate. The first part of the lab will be to create an “Any-Any-Any-Allow” rule base to test the firewall’s installation and ensure proper traffic flow. The second part will create another rule base, specifically allowing internal traffic out to the Internet.

Objectives

After completing this lab, you should have successfully:

- created a normal rule base
- created an "Any-Any-Any-Allow" rule
- saved and installed your rule
- tested your installation
- created a rule base allowing access from your network to the Internet.

Getting Started

You will create a basic security policy with the GUI. As explained in Lab 2, StoneGate allows easy control and access of all firewall functions, including the creation and installation of security policies on the firewall engines.

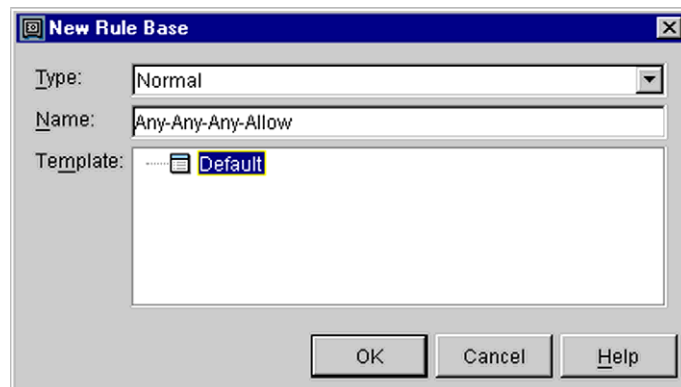
Designing a Test Rule Base



Create a Normal Rule Base

1. Click on the Security Policy Manager icon on the StoneGate Control Panel.
2. Inside the Security Policy Manager window, click on the New icon or choose **Policy** → **New** in the menus. The New Rule Base dialog box appears.
3. Select Normal from the Type selection box.
4. Name the rule base “**Any-Any-Any-Allow**” in the Name field.

ILLUSTRATION 3.1 *New Rule Base*

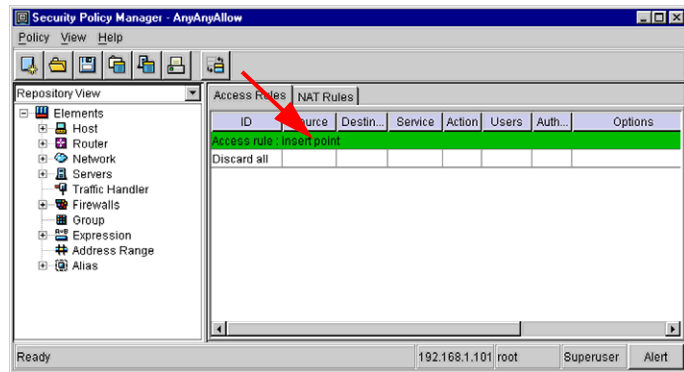


5. In the Template tree displayed below, select the Default Template.
6. Click **OK**. A normal rule base opens reproducing the content of the Default Template.

Create a Test Rule to Allow All Traffic

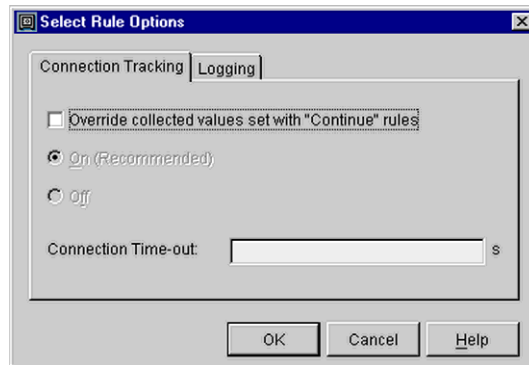
You will then add the new test rule to your rule base:

ILLUSTRATION 3.2 Access Rule Insert Point



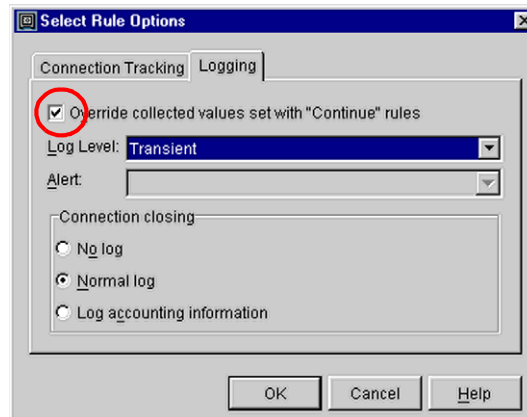
1. Add a rule by right-clicking on the green line that reads: **Access rule: insert point**, and selecting **Add Rule**. A rule appears with its ID number, default values for source, destination, service, users, and authentication each set to **NONE** or **N/A**, and a rule tag.
2. Specify the source or destination as follows:
 - Place the mouse pointer on the rule cell under the Source tab. Right click and choose **Set to ANY** from the contextual menu to match traffic from any source.
 - Repeat the previous step with the Destination tab.
 - Repeat the previous step with the Service tab.
3. Allow all traffic by left-clicking in the Action cell and selecting **Allow** from the list of options.

ILLUSTRATION 3.3 *Select Rule Options*



4. Create a log that you can test your connection with by double-clicking under the Options tab. The Select Rule Options dialog box opens as seen in Illustration 3.3.

ILLUSTRATION 3.4 *Logging settings*

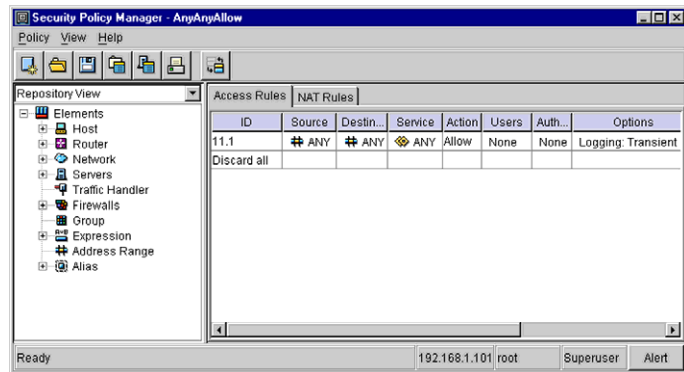


5. Switch to the Logging tab and select the check box labelled **Override collected values set with "Continue" rules**. The Log Level selection box will be enabled.
6. From the contextual menu, select **Transient**, and click **OK**.

7. Leave the radio button **Normal log** selected.

When finished, your rule should appear as follows:

ILLUSTRATION 3.5 *Any-Any-Any-Allow*



Tip:

Click the **View Inherited Rules** icon on the toolbar to see the rules inherited from the used template. These rules appear in grey background and they cannot be edited.

Save and Install the Test Rule Base on Your Firewall

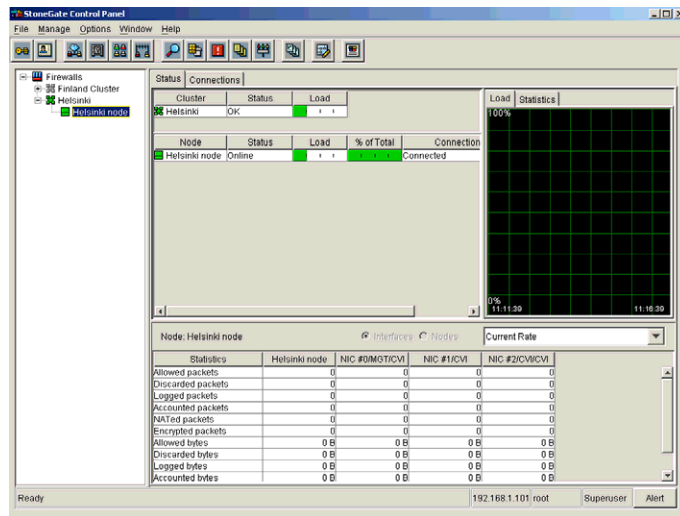
Any rule base created or updated using the Security Policy Manager needs to be saved before it can be installed on a firewall, inherited from to create other rule bases, or referenced in another rule base.

1. Click on the Save and Install icon in the Security Policy Manager toolbar.
2. From the window that appears, select the firewall that you want to install the test rule base on.
3. Wait for the installation process to complete and click **Close**.

4. Your firewall is now fully configured and ready to start processing traffic as soon as it is switched online.

Test Your Installation

ILLUSTRATION 3.6 *Node online*



1. In the StoneGate Control Panel, right-click on the icon for your firewall. Select **Go Online**. Your firewall's status should change to Online as shown in Illustration 3.6.
2. Using your Management Server, ping your ISP (212.20.x.1).
3. You should see replies to your query.
4. Open your Log Browser by clicking on the Log Browser icon. Your Log Browser will open.
5. If your test is successful, you will be able to see that connections have been allowed in your Log Browser.



Moving Ahead

Once you have successfully created and installed a test rule base, it is time to create a rule base that provides actual security. Accordingly, your next step is to create a new rule base and a rule allowing the hosts located on your internal network to access the Internet.

To create a rule allowing access to the Internet, start by creating a new normal rule base the same way you did in the previous exercise.

Create a Normal Rule Base



1. Click on the Security Policy Manager icon on the StoneGate Control Panel.
2. In the Security Policy Manager, click on the New icon or choose **Policy → New...** in the menu. The New Rule Base dialog box appears.
3. Select Normal from the Type selection box.
4. Name the rule base “**Internet Access**” in the Name field.
5. In the Template tree displayed below, select the Default Template.
6. Click **OK**. A new normal rule base opens reproducing the content of the Default Template.

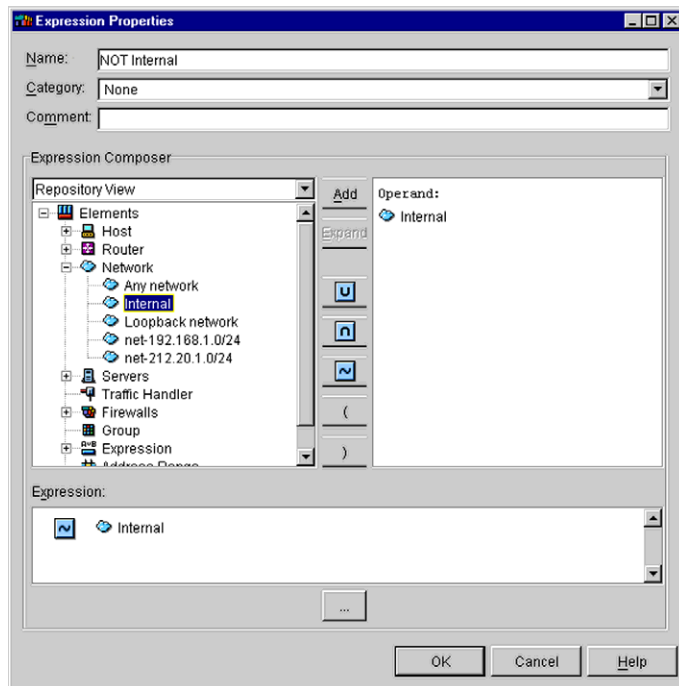
Create an Expression

In order to create a rule allowing Internet access to the internal network, you must first define an expression used to define the rule’s destination as everything but the internal network.



1. Click on the Expression icon in the toolbar. The Expression Properties window opens.

ILLUSTRATION 3.7 *Expression Properties*



2. Name your Expression “**Not Internal**” and add a comment, such as, “Anything but the internal network”.
3. Expand the Network section in the Repository View of the Expression Properties window by clicking on the + sign in front of the Network icon.
4. Select the Negation icon ~ and add it to the Expression by clicking on it once.
5. Select your internal network, it will move to the box on the right.
6. Click **Add**, and the network will drop to the bottom window behind the negate sign.
7. Click **OK**.

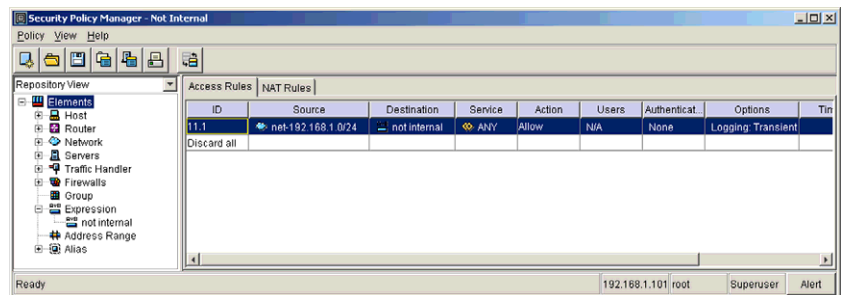
Add a New Rule Allowing Access to the Internet

Once you have created the Not Internal expression, you can add a new rule to your rule base that allows internal access to the Internet.

1. Right-click on the green line that reads: **Access rule: insert point**, and select **Add Rule**.
2. After the new rule appears, drag and drop your internal network icon, from the Repository View, into the Source cell of your new rule.
3. Drag and drop the new Not Internal Expression in the Destination cell of your new rule.
4. Right-click on the Service cell of your new rule and select ANY.
5. Click on the Action cell of your new rule and select **Allow**.
6. Double-click the Options cell of your new rule.
 - Select the Logging tab and select the check box labelled **Override collected values set with "Continue" rules**. The Log Level dialog box will open.
 - From the contextual menu, select **Transient**, and click **OK**.

Your new rule should look like the following:

ILLUSTRATION 3.8 *Not Internal Rule*



7. Save and install your new rule by clicking the Save and Install icon.

Test Your Rule Base

Test your rule base by making an HTTP connection to the Web server on the instructor's machine. To do this:

1. Type in the Address or Location field of your Web browser the IP address of your instructor's machine: **192.89.38.200**.
2. Your connection should be successful.

Summary

During this lab, you have learned how to create a normal rule base and how to save and install rule bases to your firewall. In addition, you have learned how to create expressions and how to use them in your rule base. Finally, you have experimented with StoneGate and seen how it functions once the firewall node is connected and functioning.

STOP. Wait for your instructor before continuing.

Basic Log Management

The basic log management lab will take you through simple log options and tracking. In the first part of this lab, you will examine log data from one of the rules you have already created using the Log Browser application. Then, you will use the Filtering Profile Manager to set up a profile to filter out logs generated by NetBIOS. From that point, you will examine what changes occur to the log data when you change the settings in the Options field of your rule.

Objectives

After completing this lab, you should be able to:

- create a profile, using the Filtering Profile Manager, to filter out all NetBIOS logs in your Log Browser
- set a rule's log tracking to either Alert, Transient, Essential, Stored or None, and view how the settings affect how log data is stored by using your Log Browser.

Getting Started

In Lab 3, you already created two basic rules with their logging options set to Transient. This is because you needed to be able to view the current connections in order to test your firewall's functionality, but you did not need to store log information for further use. In this lab, you will use one of these rules to expand your understanding of

filtering profiles, logging options and the differences between the various settings.



To begin the lab, you need to open your Security Policy Manager by clicking its icon on the toolbar. Select the Any-Any-Any-Allow rule base by clicking it once. Note that its Options field is set to **Logging: Transient**.

Install the Any-Any-Any-Allow rule base on your firewall by clicking the Install icon and selecting the rule base from the displayed list. Then, select your firewall from the displayed dialog box and click **OK**.

Getting Used to the Log Browser

View the Logs in Your Log Browser

Once you ensure that your site's rule base is properly installed, you may begin to view logs generated by the firewall with your Log Browser.



1. Open the Log Browser by clicking on its icon in the Launchpad.
2. You should be able to see a scrolling list of your current logs as they are generated.

ILLUSTRATION 4.1 *Log Browser*

Time	Originator	Facility	Type	Event	Action	Protocol	Src Addr	Dst Addr	Src Port	Dst Port
23.7.2002 15:48:16	Helsinki Firewall	Packet filter	Notification	New conn.	Allow	TCP (6)	192.168.1.101	192.89.38.200	1125	800
23.7.2002 15:48:31	Helsinki Firewall	Packet filter	Notification	New conn.	Allow	UDP (17)	192.168.1.101	192.89.38.200	137	137
23.7.2002 15:48:31	Helsinki Firewall	Packet filter	Notification	New conn.	Allow	ICMP (1)	192.168.1.101	192.89.38.200		
23.7.2002 15:48:31	Helsinki Firewall	Packet filter	Notification	Related p.	Allow	ICMP (1)	192.168.1.1	192.168.1.101		
23.7.2002 15:48:31	Helsinki Firewall	Packet filter	Notification	Related p.	Allow	ICMP (1)	192.168.1.1	192.168.1.101		
23.7.2002 15:48:31	Helsinki Firewall	Packet filter	Notification	Related p.	Allow	ICMP (1)	192.168.1.1	192.168.1.101		
23.7.2002 15:48:31	Helsinki Firewall	Packet filter	Notification	New conn.	Allow	UDP (17)	192.168.1.101	192.168.1.1	137	137
23.7.2002 15:48:31	Helsinki Firewall	Packet filter	Notification	Related p.	Allow	ICMP (1)	192.168.1.1	192.168.1.101		
23.7.2002 15:48:33	Helsinki Firewall	Packet filter	Notification	Related p.	Allow	ICMP (1)	192.168.1.1	192.168.1.101		
23.7.2002 15:48:34	Helsinki Firewall	Packet filter	Notification	Related p.	Allow	ICMP (1)	192.168.1.1	192.168.1.101		
23.7.2002 15:49:10	Helsinki Firewall	Packet filter	Notification	New conn.	Allow	TCP (6)	192.168.1.101	192.89.38.200	1129	800
23.7.2002 15:49:21	Helsinki Firewall	Packet filter	Notification	Connect.		UDP (17)	192.168.1.101	192.168.1.1	1125	00
23.7.2002 15:49:25	Helsinki Firewall	Packet filter	Notification	Connect.		UDP (17)	192.168.1.101	192.89.38.200	137	137
23.7.2002 15:49:37	Helsinki Firewall	Packet filter	Notification	Connect.		ICMP (1)	192.168.1.101	192.89.38.200		
23.7.2002 15:50:13	Helsinki Firewall	Packet filter	Notification	New conn.	Allow	ICMP (1)	192.168.1.101	192.89.38.200		
23.7.2002 15:50:26	Helsinki Firewall	Packet filter	Notification	New conn.	Allow	UDP (17)	192.168.1.101	192.168.1.245	138	138
23.7.2002 15:50:29	Helsinki Firewall	Packet filter	Notification	Incomplet.		TCP (6)	192.168.1.101	192.89.38.200	1129	800

LogSender 192.168.1.101, Local Time Ready 192.168.1.101 root Superuser Alert

Define a Profile that Filters out NetBIOS Logs

If you view your Log Browser, you will notice repeated logs from and to UDP ports 137 and 138. These are generated by the Windows NetBIOS protocol. By defining a specific profile, you can filter out these logs so that they no longer appear in your browser.

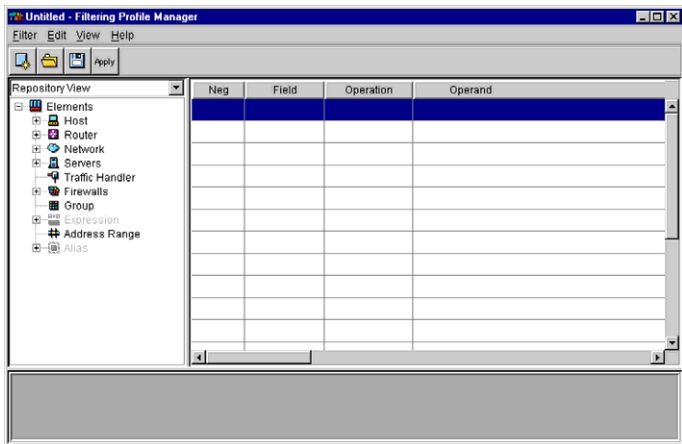


Note: In case you only want to filter out traffic, e.g., to given UDP ports—and not to TCP ports with same port numbers—you need to add another condition in the profile with protocol in the Field cell and the protocol name as the operand. The combination of these two conditions will then be applied.



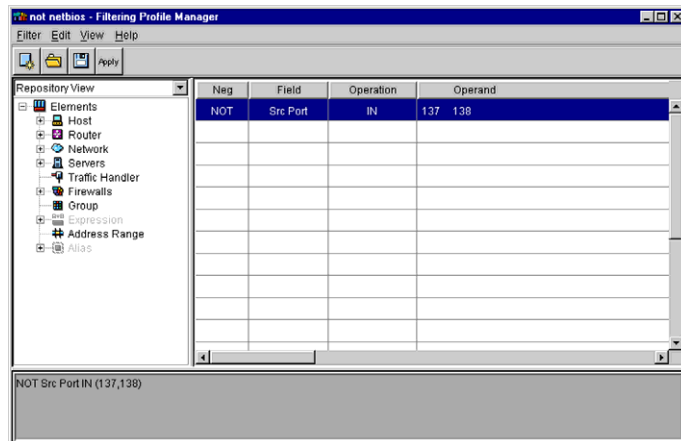
1. Ensure that your Log Browser is open.
2. Click on the **Log data filtering profile** icon to edit the filtering profile. The Filtering Profile Manager window is opened.

ILLUSTRATION 4.2 Filtering Profile Manager



3. Click on the **Neg** field and select **NOT**.

- Click on the **Field** field and select **Src Port**.
- Click on the **Operation** field and select **IN**.
- Double-click on the **Operand** field and type **137 138**. Your screen should look as follows:



You should notice that the NetBIOS logs with ports 137 and 138 are no longer being generated.

Make an FTP Connection to the FTP Server

1. Ensure that your Log Browser is open.
2. Using your Management Server, make an FTP connection to the classroom FTP server (IP address: 192.89.38.200).
 - Open the Command Prompt by clicking **Start** → **Run**.
 - Enter **ftp 192.89.38.200**, and hit ENTER.
3. View your Log Browser with the log data retrieval set to **Current**, and note the new connections that have been allowed to the FTP server.
4. Next, set the log retrieval mode to **Database**.
5. Press the HOME key on your keyboard, followed by the END key to refresh the screen. You should notice that the allowed connections are no longer visible.

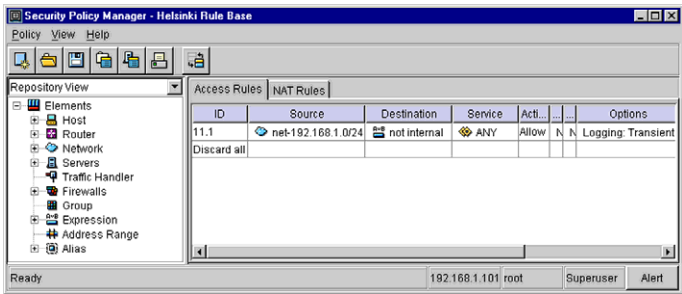


Change the Logging Option to "Stored"

In the Security Policy Manager, select the "Internet Access" rule base by right-clicking on its icon and selecting **Open**.

Once you have your rule base open, you can begin to experiment with the different log options.

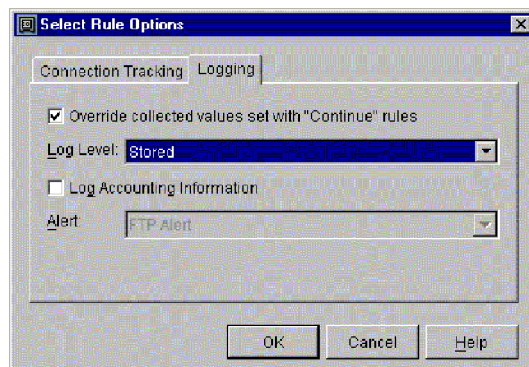
ILLUSTRATION 4.4 *Internal Internet Access Rule*



1. Double-click the Options cell of your access rule to open the Rule Options dialog box.

2. Select the Logging tab and mark the check box labelled **Override collected values set with "Continue" rules**. The Log Level field becomes active and you will see that the option is set to **Transient**.
3. From the contextual menu, change the selection to **Stored**.
4. Click **OK**.
5. Save and install the rule base on your firewall.

ILLUSTRATION 4.5 *Logging options*



Make a New FTP Connection and Check Your Log Browser

Now that you have changed the log setting to **Stored** and installed the new settings on your firewall, you are ready to create new log entries. To do this, repeat the steps for making an FTP connection and checking your logs.

1. Ensure that your Log Browser is open.
2. Using your Management Server, make a FTP connection to the classroom FTP server (IP address: 192.89.38.200).
3. View your Log Browser in the **Current** mode, and note the new connections that have been allowed to the FTP server.
4. Next, set the retrieval mode to **Database**.

5. Press the HOME key on your keyboard, followed by the END key to refresh the screen. You should notice that, this time, the allowed connections remain in your Log Browser.

Summary

In the first part of this lab, you examined log data from one of the rules you had already created using the Log Browser application. Then, you used the Filtering Profile Manager to set up a profile to filter out logs generated by NetBIOS. From that point, you examined what changes occurred to the log data when you changed the settings in the Options field of your rule.

STOP. Wait for your instructor before continuing.

Administrator Management

In this lab, you will learn basic administrator management. Specifically, you will learn how to create an additional Superuser account and how to create and modify an Operator account.

Objective

After completing this lab, you should be able to successfully:

- create a Superuser account
- create an Operator account.

Getting Started

Administrator accounts distinguish several types of administrator privileges with specific permission scopes. You can add administrator accounts when you want to delegate the management of your security policies. Depending on the permission scope of a certain account, various restrictions apply when editing simple or granted elements. The three types of administrator accounts are: Superuser, Editor, and Operator.

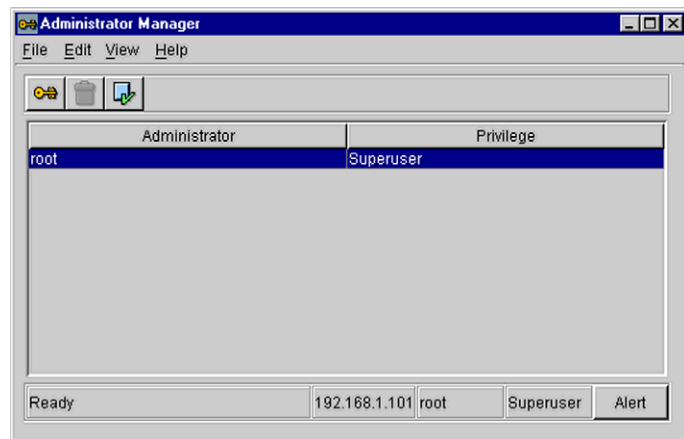
Defining Administrators

Create a Superuser Account

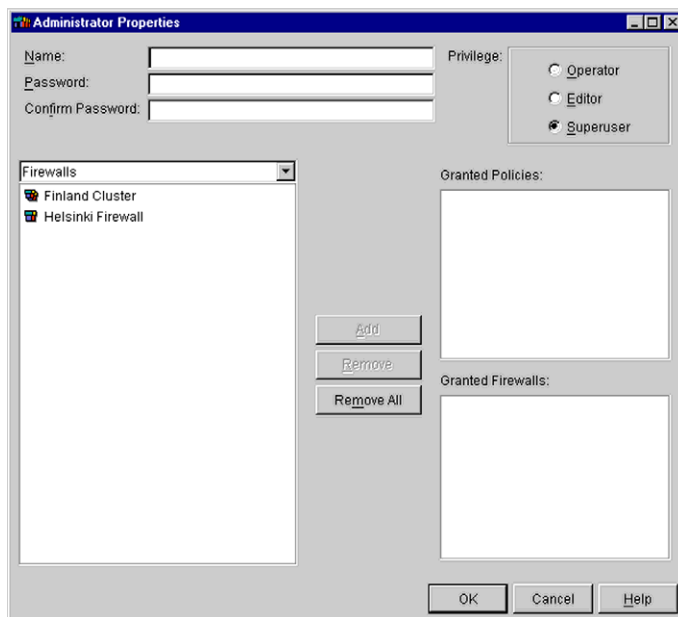


To create a Superuser account, you must first access the Administrator Manager from the StoneGate Control Panel by clicking its icon.

ILLUSTRATION 5.1 *Administrator Manager*



1. In the Administrator Manager, click the New Administrator icon on the toolbar or select **Edit** → **New** in the menu. The Administrator Properties dialog box opens as in Illustration 5.2

ILLUSTRATION 5.2 *Administrator Properties.*

2. Fill in the following fields of this dialog box:
 - **Name:** enter the user name
 - **Password:** type in the password
 - **Confirm Password:** retype the password
3. Select the Superuser privilege with the radio buttons on the right.
4. Click **OK**.

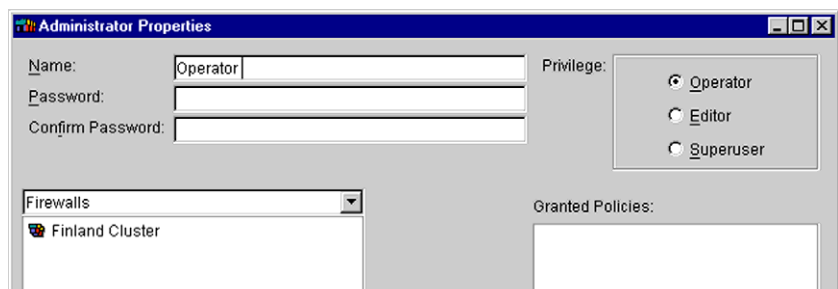


Note: *Login username is case sensitive!*

Create an Operator Account

You can create an Operator account the same way that you created the Superuser account above. Simply create a new account from the Account Manager, name it and assign a password. However, this time, select **Operator** from the radio buttons on the right. Your window should appear as follows:

ILLUSTRATION 5.3 *Administrator Properties - Operator Account*



For information regarding creating accounts and adding accounts to grant lists, see the *StoneGate Administrator's Guide*.

Checking Administrator Privileges

Test the Operator's Privileges

Once you have created the new Operator account, you can open the StoneGate Administration Client as that operator and see how StoneGate manages what actions the operator is allowed to perform. Do do so, start by closing the StoneGate Control Panel. Then, do the following:

1. Re-open the StoneGate Administrator Client, and log in as the Operator with the username and password that you defined in the previous section of this lab.
2. Open the Security Policy Manager.

3. Select the Internet access rule that you created in the previous lab, and attempt to install it on the firewall.

Grant the Operator Privileges

After you have tested the Operator's privileges in the previous section, close the Control Panel and log back in as the root Superuser again. Then, grant the Operator firewall privileges as follows:

1. Open the Administrator Manager and double-click on the Operator field. The Administrator Properties box will open.
2. From the Firewalls view, select your firewall and add it to the granted firewalls list by clicking **Add**.
3. Click **OK**.
4. Repeat the steps for testing the Operator's privileges in the above section.

Summary

In this lab you learned the basics of StoneGate administrator management. Specifically, you learned how to create additional Superuser accounts and Operator accounts. In addition, you experienced how StoneGate manages different administrator actions by checking administrator privileges.

STOP. Wait for your instructor before continuing.

Network Address Translation (NAT)

The basic network address translation (NAT) lab covers the configuration of simple NAT statements that would typically be used in a network environment.

Objectives

In the first part of the lab, you will translate the internal network with dynamic source translation behind a single external IP address 212.20.x.50

In the second part of the lab, you will translate one host from the internal network using static source translation and destination translation from 192.168.x.101 to 212.20.x.100 and back.

- First, create a static source translation rule for outbound traffic.
- Second, create a destination translation rule for inbound traffic.

Getting Started



First, if you have not already done so, log in as the Superuser to ensure that you have the correct privileges to complete this lab. Then, to continue, launch the SecurityPolicy Manager. Then, open the “Any-Any-Any-Allow” rule. When you have done this, you are ready to proceed with this lab.

Defining Dynamic NAT

In this section, you will create a dynamic source translation rule for the internal network.



.....

*Note: As with Access Rule Bases, you can view inherited NAT rules by using the **View Inherited Rules** button in the Security Policy Manager. There are two default rules for NAT, to prevent the inadvertent translation of packets between the management system components and the firewall engines.*

.....

ILLUSTRATION 6.1 NAT Rule Base

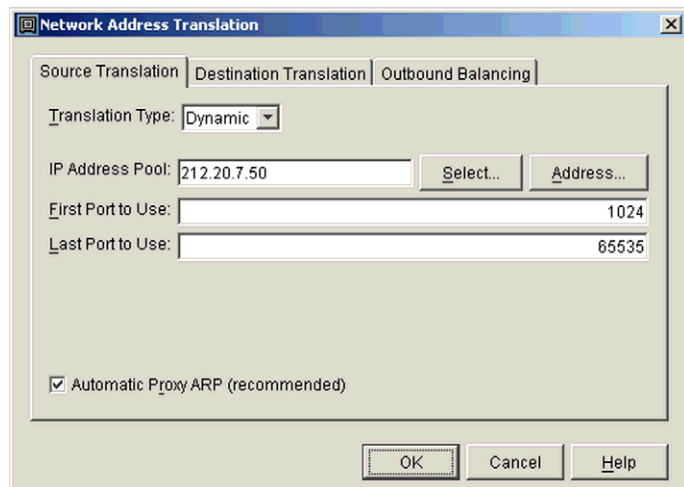
Access Rules NAT Rules					
ID	Source	Destination	Service	NAT	Used on
1	Local Cluster(NDI for managem	Log Server Management Server	SG-init SG-log		ANY
2	Management Server	Local Cluster(NDI for man	Idaps (ldap protocol ove SG-mgmt SG-monitor Idap (Lightweight Direct		ANY

Create a Dynamic NAT for the Internal Network

This dynamic NAT rule will cause all the internal IP addresses to be mapped to the address 212.20.x.50.

1. Switch to the NAT Rules tab of your 'Any-Any-Any-Allow' Rule, right-click on the green line and select **Add Rule** on the contextual menu.
2. Drag the internal network element 192.168.x.0/24 from the repository view on the left, to the Source cell of the new rule.
3. Drag and drop the **Not Internal** expression into the Destination cell of the new rule.
4. Right-click the Service cell and select **Set to ANY**.

ILLUSTRATION 6.2 *Dynamic source translation*



5. Double-click the NAT cell of the new NAT rule, and in the opened NAT dialog box, set the Translation Type to **Dynamic**.
6. Click the **Address** button and enter the value **212.20.x.50**, and click **OK**.

After you have completed these steps, the policy should look as depicted in Illustration 6.3.

ILLUSTRATION 6.3 NAT rules

Access Rules NAT Rules						
ID	Source	Destination	Service	NAT	Used on	Comme
31	net-192.168.1.0/24	Not Internal	ANY	Source: Dynamic to 212.20.1.50/32 1024-6553	ANY	
No NAT						

7. Save the policy and then install it on the firewall by clicking on the Save and Install icon.

Test the Configuration

To test that all outbound traffic is successfully translated behind the IP address 212.20.x.50:

1. Make an FTP connection to the IP address 192.89.38.200
 - Open the Command Prompt by clicking **Start** → **Run**.
 - Enter **ftp 192.89.38.200**, and hit ENTER.
2. Confirm your success by viewing the **NAT src address** in your Log Browser. You can also check the address in the connection table on the instructor's machine 192.89.38.200 using the **netstat** command.
 - Open the Command Prompt by clicking **Start** → **Run**.
 - Enter **netstat**, and click **OK**.

Adding Static NAT to the Existing Configuration

In this exercise you will create two rules: one for inbound traffic, and another for outbound traffic. Begin your static NAT configuration by defining the elements.



.....

Note: Although we are setting up both inbound and outbound rules in this lab, it is not necessary to set up pairs of translation rules for NAT. StoneGate's Multi-Layer Inspection automatically performs the reverse translation for reply packets. Accordingly, if you are only creating a rule for outbound traffic, there is no need to create an inbound rule.

.....



Configure static NAT by defining a network element for the internal machine such as a Web or FTP server. In our example we'll use the management machine for this purpose. To get started, open the Network Element Manager.



.....

Note: You could use the Management Server element for this lab, but in the interest of a real demonstration it is better to create a new element.

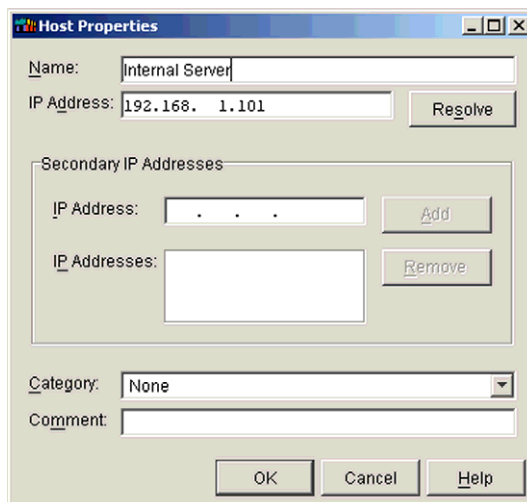
.....

Define a Network Element for the Internal Machine

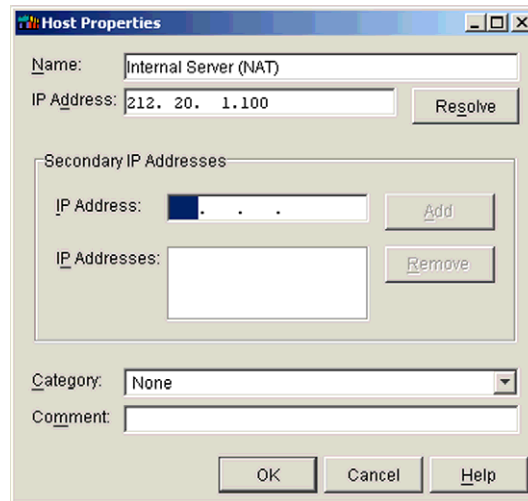


1. Select the Host icon in the toolbar to create a new host element.

ILLUSTRATION 6.4 *Host Properties*



2. In the Host properties window, name it “Internal Server” and enter the IP address **192.168.x.101** as illustrated below.
3. Create a new Host element for the NAT address in similar manner. Name it “Internal Server (NAT)” and use the public IP address **212.20.x.100** as the static NAT address.

ILLUSTRATION 6.5 *Host Properties (NAT)*

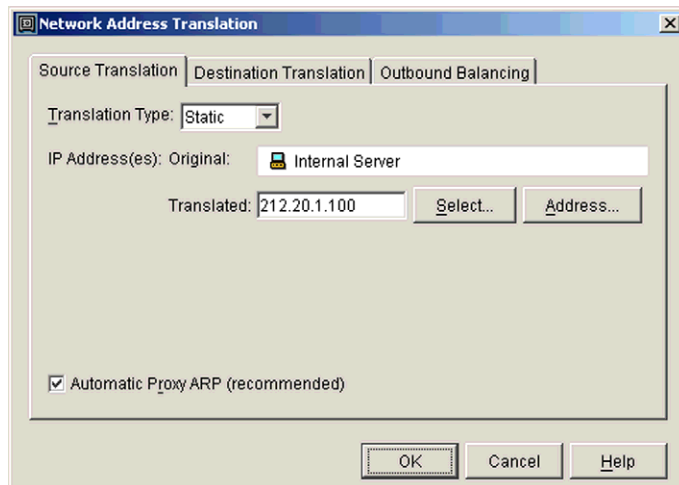
Create NAT Rules for Outbound Traffic



Having defined the elements, you will now create the rule for outbound traffic. Use the IP address 212.20.x.100 for the static NAT.

1. In the Security Policy Manager, re-open the Any-Any-Any-Allow rule base, and switch to the NAT Rules tab.
2. Right-click the dynamic NAT rule which was previously created and select **Add Rule Before**.
3. Drag the Internal Server Host element (192.168.x.101) you created from the Repository View to the Source cell of the new rule.
4. Drag the **Not Internal** expression to the Destination cell of the new rule.
5. Right-click the Service cell and select **Set to ANY**.

ILLUSTRATION 6.6 *Static source NAT*

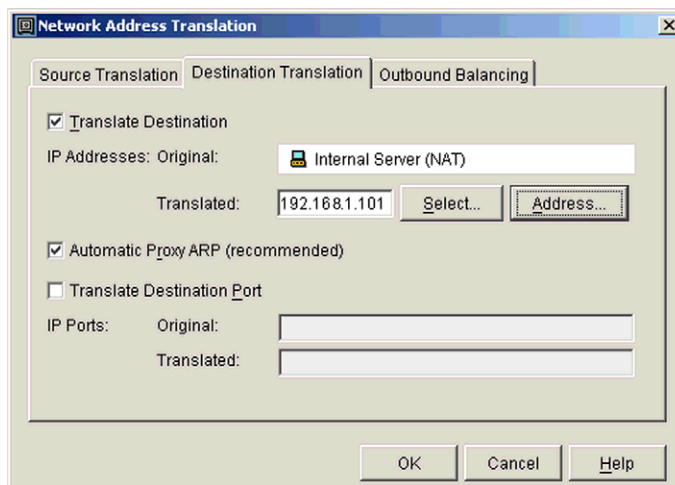


6. Double-click the NAT cell and, in the opened NAT dialog box, set the Translation Type to **Static**.
7. Click the **Address** button in the NAT dialog box and enter the value **212.20.x.100**, and click **OK**.

Define the NAT Rule for Inbound Traffic.

You will now add the NAT rule for inbound traffic. Translate the destination address 212.20.x.100 to 192.168.x.101.

1. Right-click the last rule created and select **Add Rule After**.
2. Drag the **Not Internal** expression into the Source cell of the new rule.
3. Drag the Internal Server's NAT Host element (212.20.x.100) from the Repository View to the Destination cell.
4. Right-click the Service cell and select **Set to ANY**.

ILLUSTRATION 6.7 *Destination Translation*

5. Double-click the NAT cell of the new rule to open the NAT dialog box. In the Destination Translation tab, select the Translate Destination check box.
6. Click the **Address** button in the NAT dialog box and enter the value **192.168.x.101**, and click **OK**.

The completed rule base should look as depicted in Illustration 6.8.

ILLUSTRATION 6.8 *NAT rule base*

Access Rules		NAT Rules				
ID	Source	Destination	Service	NAT	Used on	Comr
3.1	Internal Server	Not Internal	ANY	Source: Static from Internal Server	ANY	
3.2	Not Internal	Internal Server (NAT)	ANY	Destination: Static from Internal Server	ANY	
3.3	net-192.168.1.0/24	ANY	ANY	Source: Dynamic to 212.20.1.50/32	ANY	

7. Save the policy and install it on the firewall by clicking on the Save and Install icon.

Test the Configuration

Test that all outbound traffic is translated behind the IP address 212.20.x.100.

1. Make an FTP connection to the instructor's machine 192.89.38.200.
 - Open the Command Prompt by clicking **Start → Run**.
 - Enter **ftp 192.89.38.200**, and click **OK**.
2. Confirm your success in the connection table on the 192.89.38.200 machine using the `netstat -an` command.
 - Open the Command Prompt by clicking **Start → Run**.
 - Enter **netstat -an**, and click **OK**.

Summary

In the first part of the lab, you translated the internal network with dynamic source translation behind a single external IP address 212.20.x.50

In the second part of the lab, you translated one host from the internal network using static source translation and destination translation from 192.168.x.101 to 212.20.x.100 and back.

STOP. Wait for your instructor before continuing.

Basic User Authentication

In this lab exercise, you will create a group and a number of user accounts in the internal directory services and a rule using simple password authentication for these users.

Objectives

After completing this lab exercise, you will have:

- configured a group and a few user accounts in the StoneGate internal directory services
- designed a rule using user authentication with simple passwords
- developed an understanding regarding how StoneGate performs user authentication based on its internal directory.

Getting Started

In this exercise we shall focus on simple password authentication which relies on StoneGate's internal user information. External authentication methods shall be explained in the *Advanced Implementation and Beyond* course.

StoneGate Management Server stores user and group definitions in its internal database. Each firewall engine runs a local, identical replica of this database, and any updates to the main user database are automatically and immediately propagated to the nodes. The nodes



handle any authentication tasks on the basis of their internal database replicas. First, however, you must create the group and user entries to the database. In the User Manager, the group and user accounts are created under the default StoneGate domain (InternalDomain) and the default parent group.

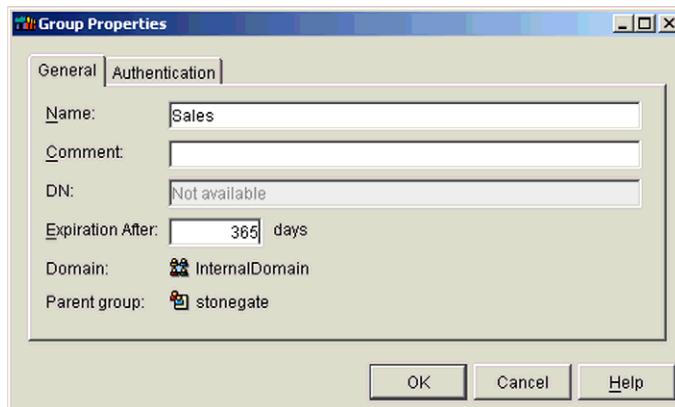
Defining User Accounts

Create a Group of Users



1. Open the User Manager.
2. In the User Manager, in the User Directory tab, you can see **InternalDomain**, and the default parent group under that. Right-click the default group, and select **New** → **Group** from the contextual menu.

ILLUSTRATION 7.1 *Group Properties*



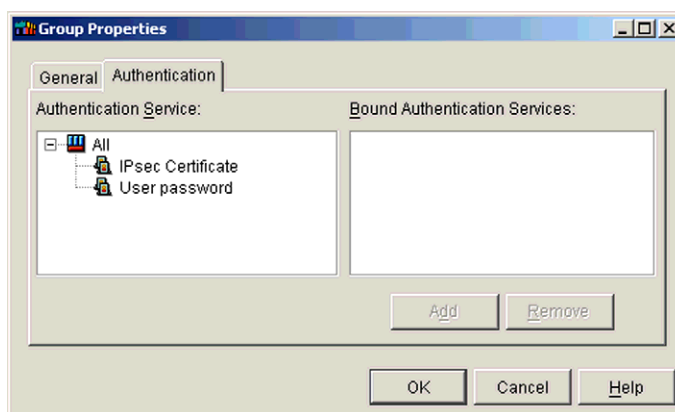
3. In the displayed Group Properties window, enter the group name and add a comment in the next field, if you like.



*Note: The **DN** (Distinguished Name) parameter is not user definable in case of internal StoneGate authentication.*

4. Set the expiration time for the group.

ILLUSTRATION 7.2 *Authentication for the group*



5. Switch to the Authentication tab, and define the authentication method for the group. In this case, select the default **User Password** from the left panel, and click **Add** to make it the bound authentication service for this specific group of users.
6. Click **OK**.

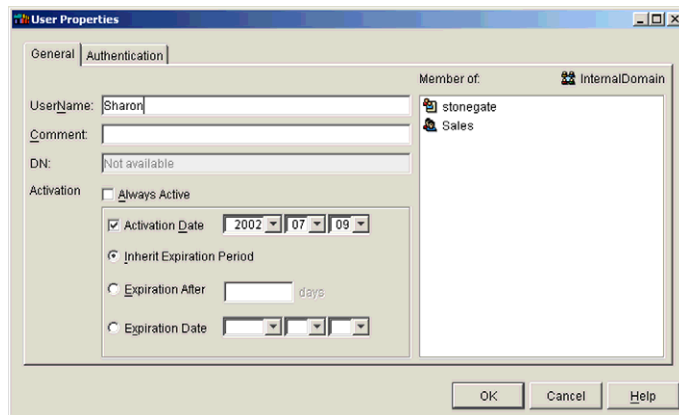
Create User Accounts

You can create user accounts both under the parent group and under the group you just created. By using groups, you can collect those users that should be treated similarly as regards authentication under one element.



1. In the User Directory tab, you can see **InternalDomain**, and the default parent group under that. Click the + sign next to the default group to see all the groups under it. Right-click the group you just created, and select **New** → **User** from the contextual menu to create a user account that is a member of that group.

ILLUSTRATION 7.3 *User Properties window; General tab*

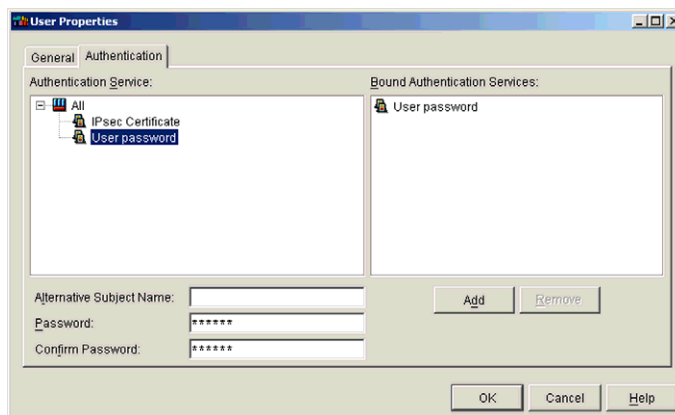


2. In the displayed User Properties window, enter the user ID of the user in the **UserName** field, and add a comment in the next field, if you like.



*Note: The **DN** (Distinguished Name) parameter is not user definable in case of internal StoneGate authentication.*

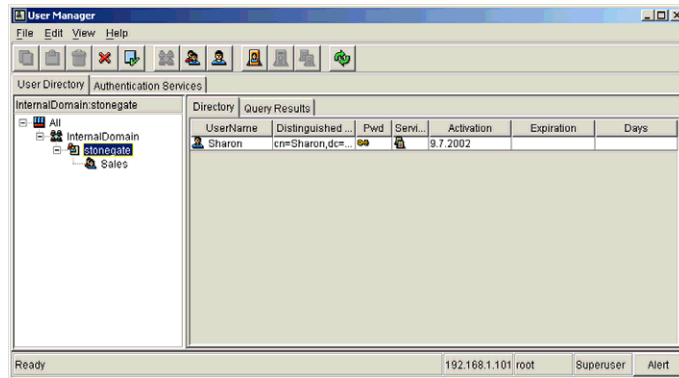
3. Next, define the activation of the user account. Set the activation and expiration of the user account. Try different activation parameters for the different accounts you'll create.

ILLUSTRATION 7.4 *User Properties window, Authentication tab*

4. Switch to the Authentication tab, and define the authentication method for the user. In this case, select **User Password** from the left panel, and click **Add** to make it the bound authentication service for this specific user.
5. Define the password for the user, and confirm it in the two fields at the bottom of the window.

Now you have defined a StoneGate user account using simple password for authentication. You can proceed with creating more users. The created users with their parameters are displayed in the User Manager main window (see Figure 7.5).

ILLUSTRATION 7.5 User Manager main window



.....

Tip: If you need to change a user's password, you can simply select the user account from the list displayed in the User Manager main window, and then right-click and select **Change Password**.

.....

Designing and Testing Authentication Rules

Create an Authentication Rule

After you have defined a number of user accounts, you are ready to design a rule that will require simple password authentication for a specified user. The rule shall allow outbound HTTP connections with client IP authentication for certain users only.







1. In the Security Policy Manager, create a new rule base by clicking the New icon on the toolbar.
2. In the opened dialog box, select **Normal** as type, name the rule base **Authentication**, and select the **Default** as its template.
3. Click **OK** to open the Editor window for the new rule base.

4. Insert a new rule by clicking the green row stating **Access rule: insert point**.
5. Fill in the cells as follows:
 - Source: your internal network (192.168.x.0)
 - Destination: the **Not Internal** expression
 - Service: **HTTP**
 - Action: **Allow**.
 - User: drag and drop a user or a whole group you have created from the User View.
 - Authentication: drag and drop **User Password** from the Authentication Service View.
 - Options: Log level **Stored**.
6. Double-click the Authentication Service object in the Authentication cell.
 - In the opened Authentication Parameters window, select **Require Authentication** for method and **Client IP** for authorization.
 - Click **OK**.

Your authentication rule should now look as depicted in Illustration 7.6.

ILLUSTRATION 7.6 *Authentication rule*

Access Rules		NAT Rules						
ID	Source	Destination	Service	Action	Users	Authenticat.	Options	Time
12.1	 net-192.168.1.0/24	 Not Internal	 http (Vo)Allow		 Sales	Authorize cli Client initiat time-out = 3f	Logging: Stored	
Discard all								

7. Click the Save and Install icon to install the security policy on your firewall cluster.

Test User Authentication

Now that the authentication rule has been included in the current security policy, you can test how StoneGate performs authentication for a user registered in the internal user database. In case of client initiated authentication you should first connect to the firewall to authenticate yourself using a valid user name and password combination. The contacted StoneGate firewall engine will then compare the entered information with its internal database to decide whether the connection request can be accepted or not.

Test an HTTP connection:

1. Open location **192.89.38.200** with your Web browser.
2. Did it work? If not, why? Discuss this with your instructor.

Make a telnet connection to the firewall:

Open a telnet connection to your StoneGate firewall internal interface, port 2543.

1. Select **Start** → **Run**, and enter **telnet** and the internal IP address and port of your firewall node (**192.168.x.1 2543**) on the command line.
2. Once the telnet window opens, enter the username and password combination for a user which you have created.
3. If your username-password combination was correct, your client IP address is now authorized by StoneGate.



.....
Tip: Test also using an invalid username-password combination.
.....



.....
Note: Telnet is not actually allowed through StoneGate.
.....

Test an HTTP connection again:

1. Again, open location **192.89.38.200** with your Web browser.
2. Did it work this time?



.....
Tip: Keep your Log Browser open, so that you can follow how performing these tests is reflected in the log data.
.....

Summary

During this lab you created a number of user accounts in the internal directory services and a rule using simple password authentication. External authentication methods are covered in the *Advanced Implementation and Beyond* course.

STOP. Wait for your instructor before continuing.

VPN Fundamentals

During the Virtual Private Network (VPN) Fundamentals lab, you will create a StoneGate-to-StoneGate single firewall VPN. You will first define two different cities as gateways and bundle the internal networks behind those gateways. Then, you will create the VPN between the sites and use an arbitrary, pre-shared key as the secret key information. Using the Log Browser, you will monitor an outbound connection between the gateways. This lab illustrates that VPN settings permit data flow in a single direction. For VPN traffic to flow from one gateway to the other, sites in both the source and the destination fields must be separately configured to allow two-way traffic.

Objectives

After completing this lab, you should have successfully:

- defined two security gateways
- created and configured a VPN element
- created two VPN rules
- tested your VPN.

Getting Started

To get started, you will first need to create the network elements acting as the VPN end-points; that is, your and your neighbor's StoneGate security gateway. Then, you will be able to build and configure a VPN between them using the VPN Manager.

Defining the Network Elements

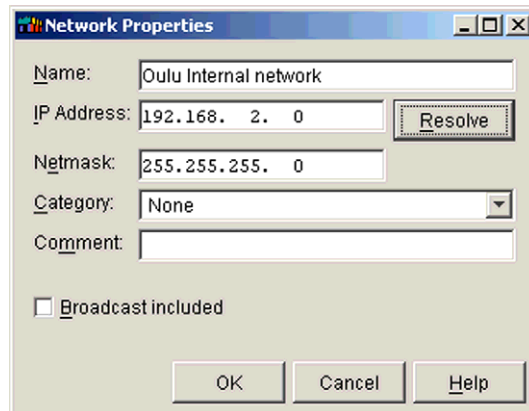
Create a Network Element for Your Partner's City

Start building the VPN by defining the Network of your partner's city in the Network Element Manager. Your own internal network is defined already.



1. In the Control Panel, open the Network Element Manager by clicking its icon on the Launchpad.
2. Click the Network icon on the toolbar to create a new Network.

ILLUSTRATION 8.1 *Network Properties*



3. In the opened Network Properties dialog box, enter your partner's internal network's name (e.g., "Oulu Internal Network").

4. Enter the internal network's IP address (**192.168.x.0**) and the netmask (**255.255.255.0**).
5. Click **OK**.

Create the Internal Security Gateway



Next, create your internally managed security gateway.



1. In the StoneGate Control Panel, open the VPN Manager by clicking its icon on the Launchpad.
2. Click the Internal Security Gateway icon in the toolbar.

ILLUSTRATION 8.2 *Internal Security Gateway General Properties*

Internal Security Gateway Properties

General | End-Points | Capabilities | Trusted CAs

Security Gateway

Name:

Associated Firewall Cluster

Firewall:

VPN Client NAT Pool

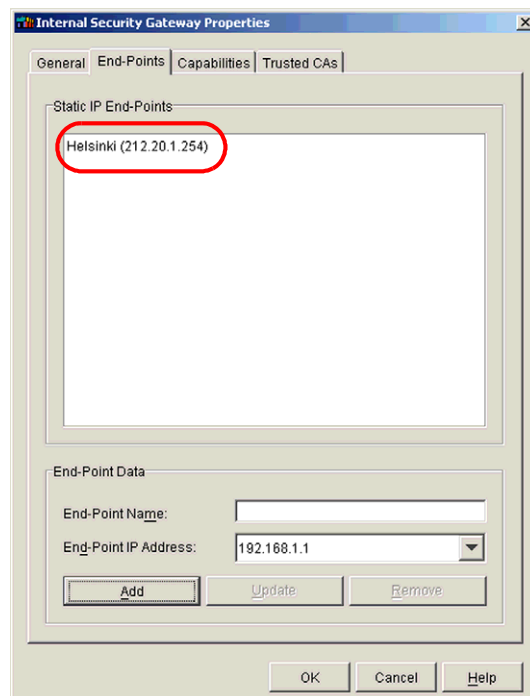
IP Range: . . to . . .

Port Range: to

3. In the General tab of the Internal Security Gateway Properties dialog box, name your gateway (e.g., “Helsinki SGW”).

4. Select your local firewall from the options provided. The default **SGW Settings** in the other tab needn't be changed.
5. The **VPN Client NAT Pool** will be left blank. The use of VPN client NAT Pool will be covered in the *StoneGate Advanced Implementation and Beyond* course.

ILLUSTRATION 8.3 *Internal Security Gateway End-points Properties*



6. Switch to the End-points Tab and then name the end points.
7. Select your firewall's external IP address, and click **Add** to insert the name and IP address of the end-point in the text box.
8. Click **OK**.

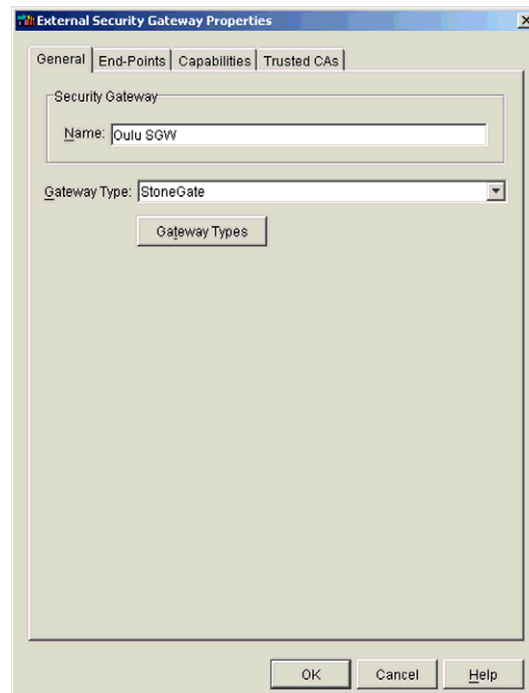
Create the External Security Gateway

You need to define the other end of the VPN next. Therefore, you must create also your partner's security gateway as element.



1. In the VPN Manager, click the External Security Gateway icon to open the External Security Gateway Properties dialog box.

ILLUSTRATION 8.4 *External SGW General Properties*



2. In the General tab, name the external gateway by entering the name of the security gateway of your partner's city (e.g., "Oulu").
3. Select **StoneGate** as the **Gateway Type**. The **Gateway Types** needn't be changed.



Tip: Remember that external in this sense refers to an externally managed security gateway; not to the location of the other gateway. In other words, you can also create internal-to-internal SGW VPNs if all gateways are managed by the same StoneGate management system.

ILLUSTRATION 8.5 External SGW End-points Properties

4. Switch to the End-points tab, click the radio button **Static IP**.
5. Give the end-point the name of your partner's city and select its external IP address (**212.20.x.254**).

6. Click the **Add** button to insert the name and IP address of the endpoint in the text box.
7. Click **OK**.

Configure the Encryption Domains

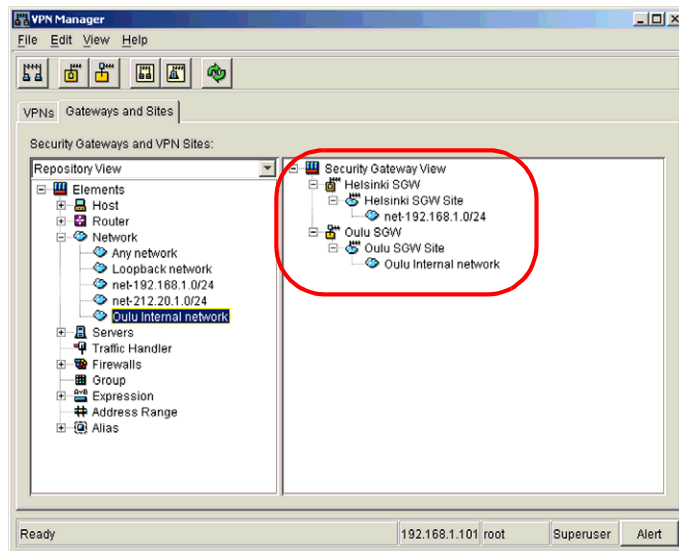
You need to assign sites to both defined security gateways. In this exercise, you can assign the internal network to the security gateways. Start with your internal network and your own security gateway.

1. In the VPN Manager, select the Gateways and Sites tab. Ensure that you have the Repository View on the left panel.
2. Expand the Network elements by clicking the + sign.
3. Drag and drop your internal network from the left onto your internal security gateway on the right panel.

You need to repeat the previous step also for the external security gateway.

4. Drag and drop your partner's internal network from the left onto the external security gateway on the right panel.
5. When finished, your Security Gateway View should resemble Illustration 8.6.

ILLUSTRATION 8.6 *Security Gateway View*



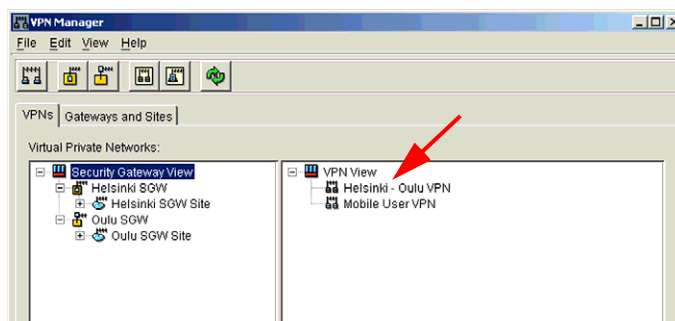
Create a VPN Element

After defining the security gateways functioning as end-points of the VPN, you can create the actual VPN element.



1. In the VPN Manager, click the VPN icon.
2. In the displayed dialog box, specify the name of the VPN. Use the names of the two cities (e.g., “Helsinki - Oulu VPN”). Click **OK**.

ILLUSTRATION 8.7 VPN View



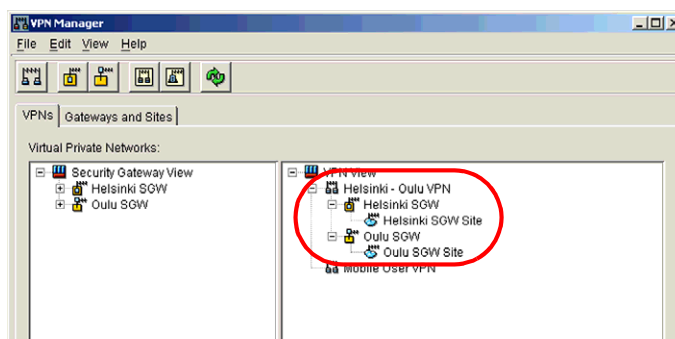
3. Switch to the VPNs tab to see the newly created VPN element (as in Illustration 8.7).

Setting up the VPN

Configure Your VPN

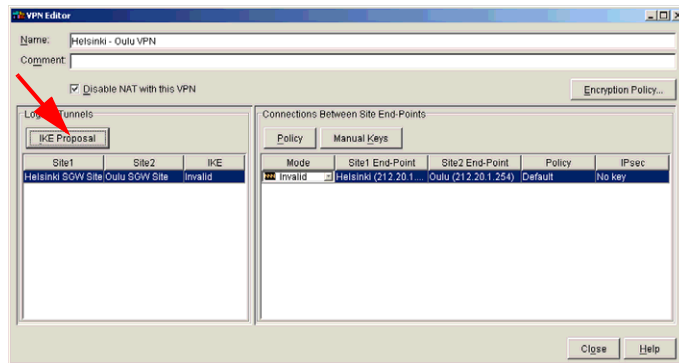
Now that you have created a VPN, you need to configure its settings.

ILLUSTRATION 8.8 VPN View

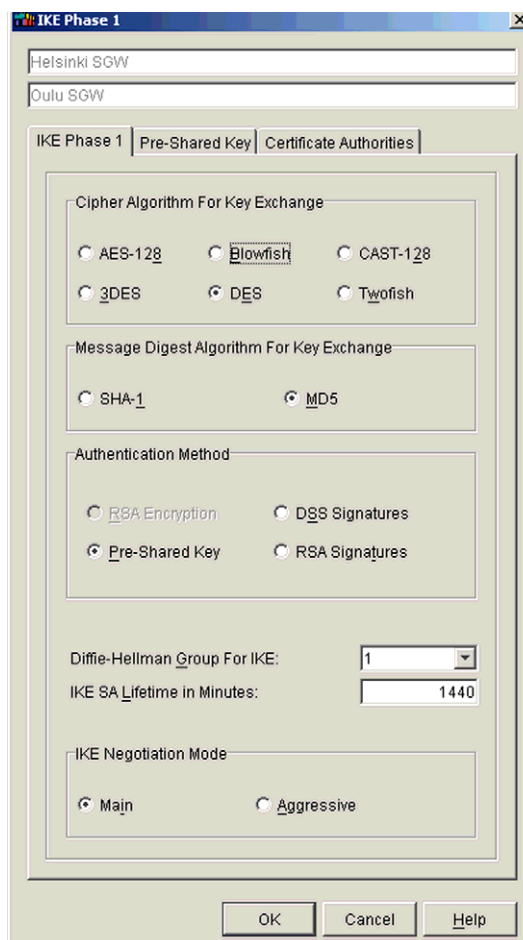


1. In the VPNs window, drag and drop both gateway elements from the left panel to right onto the VPN element you created.

ILLUSTRATION 8.9 *VPN Editor*

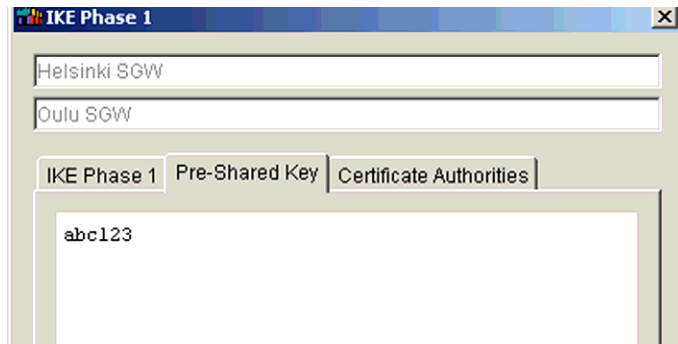


2. Set the properties of the VPN by selecting the VPN you just created. Right-click on it and select **Properties** from the contextual menu. The VPN Editor window will open.
3. In the VPN Editor window, click on the **IKE proposal** button located in the Logical Tunnels panel on the left. The IKE Phase 1 window will open. See Illustration 8.10.

ILLUSTRATION 8.10 *IKE Phase 1*

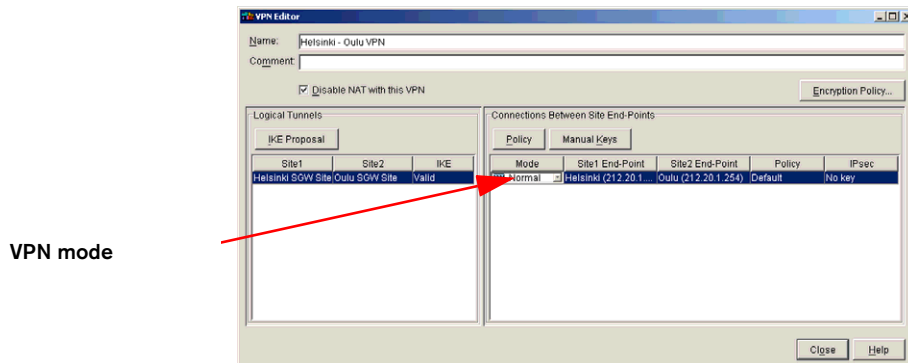
4. In the displayed IKE Phase 1 dialog box, select the IKE Phase 1 tab. Set the **Diffie-Hellman Group for IKE** to the value 1. Accept all the other defaults.

ILLUSTRATION 8.11 *Pre-shared key*



5. Switch to the **Pre-Shared Key** tab.
6. Type in an arbitrary key used for authentication in this exercise, such as “abc123.” The Certificate Authorities tab needn’t be changed.
7. Click **OK** to return to the VPN Properties dialog box..

ILLUSTRATION 8.12 *Connections between Site End-points*



VPN mode

8. Under the Connections Between Site End-Points header on the right panel, click the first selection box in the Mode column. Change the setting from **Disabled** to **Normal**.
9. Click **Close**.



Tip: *When you move the mouse over the Mode cell, a text box containing the VPN settings will be displayed.*

Designing and Testing VPN Rules

Create a VPN Rule Base



After you have configured the VPN between the two gateways, you can create access rules to test how VPN traffic is handled by StoneGate. Open the Security Policy Manager to design the rules.

1. Create a new policy by clicking the New icon on the toolbar.
2. In the opened dialog box, set the type as Normal, name the rule base as “VPN”, and select Template as default.
3. Once your new rule base opens, click on the green line saying **Access rule: insert point**, and click **Add Rule**.
4. For the new rule, fill in the cells as follows:
 - Source: drag and drop the internal network here.
 - Destination: drag and drop your partner’s network here.
 - Service: **FTP, HTTP**.
 - Action: select **Enforce VPN**, and select your VPN from the options provided.
 - Options: set the Log Level as **Stored**.
5. Create a new rule under the one you just created by right-clicking on its row and selecting **Add Rule After**.
6. Fill in the cells as follows:
 - Source: drag and drop your partner’s network here.
 - Destination: drag and drop your internal network here.
 - Service: **FTP, HTTP**.

- Action: select **Enforce VPN**, and select your VPN from the options provided.
 - Options: set the Log Level as **Stored**.
7. Save and install the policy by clicking the Save and Install icon.

ILLUSTRATION 8.13 VPN rules

ID	Source	Destination	Service	Action	Users	Auth...	Options	Time
12.1	net-192.1...	net-212...	ftp (File Transfer Protocol)	Enforce VPN Helsinki - Oulu VPN	N/A	None	Logging: Stored	
12.2	net-212.2...	net-192...	http (Web)	Enforce VPN Helsinki - Oulu VPN	N/A	None	Logging: Stored	
Discard all								

Test Your VPN

Now that you have created a VPN and the VPN rules, you can test how StoneGate handles VPN traffic.



1. Open the Log Browser, so that you can inspect how VPN traffic is reflected in the log data.
2. Open an FTP connection to the IP address of your partner's Management Server (**192.168.x.101**).
3. In the Log Browser, check that the VPN connection works.



Tip: Drag the Info Message column of the Log Browser to the right so that you can view it simultaneously with the source and destination address information. The column contains information on how the VPN connections are being established.



APPENDICES



Glossary

A

Abuse of Privilege

When a user performs an action that they should not have, as defined by the organizational policies or by law.

Access Control

Restricting or allowing access to specific resources based on pre-defined criteria. In StoneGate, this means the part of the security policy that defines access to resources based on authentication.

ACPI (Advanced Configuration and Power Interface)

Please see “*Advanced Configuration and Power Interface (ACPI)*” on page 292.

Action

What the firewall engine should do with a packet that matches the criteria for a particular rule in the security policy. The Action can be:

- **Allow** - the packet is sent through the firewall
- **Discard** - the packet is discarded by the firewall
- **Refuse** - the packet is discarded and an ICMP error packet is returned to the source

- **Jump *sub*** - the firewall engine evaluates the packet using the sub-rule base *sub*
- **Enforce VPN** - the firewall will allow the connection if the specified VPN is used, otherwise the connection is discarded
- **Apply VPN**- the firewall will allow the connection if the specified VPN is used. If the specified VPN is not used, the rule is no longer considered a match, and rule base traversal continues
- **Continue** - the firewall processes the specified Rule Options for the packet and continues traversing the rule base (Please see “*Rule Option*” on page 320.)

Please see “*Rule*” on page 320.

Address Resolution Protocol (ARP)

An Internet standard (RFC 826) protocol used to associate IP addresses with the media hardware address of a network interface card on a local area network (LAN).

Advanced Configuration and Power Interface (ACPI)

A specification for power management developed by Microsoft, Intel, and Toshiba. ACPI is a replacement standard for the older Advanced Power Management (APM) specification.

Aggressive Mode

The authentication of two IPsec end-points with only three messages, as opposed to Main Mode’s six. Aggressive mode also does not provide PFS support, and SA negotiation is limited. Please see “*Main Mode*” on page 310. See also “*Security Association (SA)*” on page 321.

AH (Authentication Header)

Please see “*Authentication Header (AH)*” on page 294.

Anti-Spoofing

A technique used to protect against IP spoofing. Please see “*IP Spoofing*” on page 307.

API (Application Programming Interface)

A set of functions, syntax or languages by which programs can access and use operating system components, application components, or network services.

APM (Advanced Power Management)

Please see “*Advanced Configuration and Power Interface (ACPI)*” on page 292.

Application Layer Gateway; Application Level Firewall

A firewall system, or gateway, in which packets are examined based on the application protocol being used (e.g., telnet, FTP, SMTP). Proxies for each application-level service are installed on the gateway, and are often configured to relay a conversation between two systems. That is, a packet’s destination is the gateway, which then establishes a separate connection to the other system to complete the connection.

ARP (Address Resolution Protocol)

Please see “*Address Resolution Protocol (ARP)*” on page 292.

Asymmetric Encryption

A cryptography technology that uses a pair of matched keys. Encryption is performed with the public half of a pair and only the matching private half can perform decryption. Private keys are secret while public keys are available to any party that wants to communicate with the holder of a private key. Public key technology can be used to create digital signatures and deal with key management issues. It is also referred as public key encryption. Please

see “*Symmetric Encryption*” on page 325. See also “*Public Key Cryptography*” on page 317.

Authentication

The process of proving that someone or something is who or what they claim to be.

Authentication Header (AH)

A security protocol supported by the IPsec protocol to enhance traffic security. It enables the authentication and integrity of data against packet corruption or tampering. AH protocol can use SHA-1 or MD5 to generate a hash signature based on a secret component from the SA, the packet payload and some parts of the packet header. Please see “*Security Association (SA)*” on page 321.

Authentication, pre-shared key

An authentication mechanism whereby the key used in encryption is exchanged beforehand.

Authentication Token/Authenticator

A portable device for authenticating a user. Authentication tokens typically operate by challenge/response, time-based code sequences, or other techniques.

Authorization

Process of giving someone/something permission to do or have something. Usually related to authentication; once a user has authenticated (proved who they are), they are authorized (given permission) to perform certain actions.

Autonomous System

An internetwork in which interior routing protocols are used to control routing.

B

Bastion Host

A system that has been hardened to resist attack, and which is installed on a network. It is a system that is expected to come under attack. They are often components of firewalls, or public systems (e.g., Web servers, anonymous FTP servers).

Block Cipher

Please see “*Bulk Encryption Algorithm*” on page 295.

Blowfish

A bulk encryption algorithm that uses 64-bit blocks, and variable-length keys from 32 to 448 bits, designed by Bruce Schneider.

Border Routing

Used for connections between different autonomous systems.

Bulk Encryption Algorithm

Describes symmetric encryption algorithms which operate on fixed-size blocks of plaintext and generates a block of ciphertext for each. DES is one of the best known and widely used algorithms, but it is now obsolete. The current generation of bulk algorithms -- such as Blowfish, CAST-128, and IDEA -- all use 64-bit blocks. The next generation, AES, uses 128-bit blocks and supports key sizes up to 256 bits. Please see “*Stream Encryption Algorithm*” on page 325.

C

CA (Certificate Authority)

A trusted third-party organization or company that issues digital certificates, used to create digital signatures and public-private key pairs. The role of the CA in this process is to guarantee that the

individual granted the unique certificate is, in fact, who he or she claims to be.

CAST-128

A bulk encryption algorithm using 64 bit blocks and keys from 40 bits to 128 bits in size, although keys smaller than 128 bits are padded. It is described in detail in RFC 2144, an informational RFC upon which several proposed standards are based.

Certificate, Digital

An electronic identification card for a user or device. Digital certificates are distributed, or granted, by certificate authorities (CAs), and ensure that the user or device is who/what they claim to be. Digital certificate holders have a public and private key pair, which can be used to sign messages (authenticate the sender), and decrypting incoming messages (ensuring only the certificate holder can decode the encrypted message).

Challenge/Response

An authentication technique whereby a server sends an unpredictable challenge to the user, who computes a response using some form of authentication token, which can be an authenticator, or pre-shared keys used to encrypt random data.

Ciphertext

The encrypted, or enciphered, form of text, resulting from some having some cryptographic function applied to the original plaintext. Please see “*Cryptography*” on page 298.

Client

In a client-server architecture, it is usually an application running on a computer or a workstation and using a server for some operations.

Cluster

A group of devices, or nodes, that share a given work load. A fast network connection between the nodes is typically needed because of the communication requirements between devices.

Clustering Technology

A set of methods and algorithms used to implement highly scalable solutions where more than one machine handles the work load. The advantages of clustering technology include increased performance, availability, and reliability.

Cluster Virtual Interface (CVI)

A logical interface shared by all nodes in a cluster. A CVI is assigned an IP and MAC address, which are then used by every node in a cluster for communication. These interfaces give the cluster a single identity on the network, reducing the complexity of routing and network design.

Clustered Multi-Link VPN

Clustered Multi-Link Virtual Private Network. Technology that uses a cluster of VPN devices and several network connections to form a single, logical VPN. Clustered Multi-Link VPNs offer increased performance and high availability where neither a single VPN device nor an individual network link becomes a single point of failure. See also “*Multi-Link Technology*” on page 312.

Connection Identification

A component of multi-layer inspection technology that uniquely identifies a connection. Please see “*Multi-Layer Inspection*” on page 312.

Connection Tracking

The set of data maintained for a connection. Used for relating incoming packets to existing connections. Connection tracking

information also includes information to support features like NAT, Load Balanced Routing and Protocol Agents. May also contain accounting information.

Content Filtering

Activity that modifies or stops the flow of certain connections according to the examination of data contents. Common examples include virus scanning or filtering of Web URLs. Also known as *content inspection* or *content screening*.

Content Inspection

A method that enables additional third-party services, such as anti-virus scanners, to perform a more detailed examination of a connection's data, and assisting in the determination to allow or discard the packets.

Cryptography

The art of protecting information by transforming plaintext (encrypting it) into an unreadable format, called ciphertext. Only those who possess a secret key can decipher (or decrypt) the message into plaintext.

Cryptographic Checksum

A one-way function applied to a file to produce a unique “fingerprint” of the file for later reference. File tampering can then be discovered by verifying the checksum value in the future.

CVI (Cluster Virtual Interface)

Please see “*Cluster Virtual Interface (CVI)*” on page 297.

D

Data-Driven Attack

A form of attack in which the attack is encoded in innocuous-seeming data. The data is then inadvertently executed by a user or software to implement an attack. Because firewalls typically do not examine the data portion of packets, data driven attacks can bypass firewall software.

DES (Data Encryption Standard)

A symmetric key bulk encryption algorithm that uses 56-bit keys. DES is now considered a weak encryption, vulnerable to brute force attacks.

3DES (Triple Data Encryption Standard)

A symmetric key bulk encryption algorithm that improves on the relative weakness of DES by processing the data three times with three different 56-bit keys.

DHCP (Dynamic Host Configuration Protocol)

Please see “*Dynamic Host Configuration Protocol (DHCP)*” on page 300.

Diffie-Hellman (DH)

A key exchange algorithm using various key sizes (theoretically unlimited) to create a shared secret key. It compliments RSA and DSA algorithms in the IPsec protocol, but is purely a key exchange algorithm, and cannot be used to encrypt data.

Digital Certificate

Please see “*Certificate, Digital*” on page 296.

DMZ Network

DeMilitarized Zone Network. A network separate from both internal and external networks, and connected through a gateway. Often used for isolating bastion hosts or publicly available machines, e.g., mail and HTTP servers are typically located in a DMZ network.

DNS Spoofing

An attack method whereby the DNS name of a system is assumed by a malicious system, either by corrupting the name service cache of a victim, or by compromising a domain name server for a valid domain. The victim system is then directed to the malicious system instead of the original server.

Dual-Homed Gateway

A system with two or more network interfaces, each of which is connected to different networks. Firewalls are a typical configuration of a dual-homed gateway, performing filtering and blocking traffic passing between the networks.

Dynamic Host Configuration Protocol (DHCP)

A protocol for dynamically assigning IP addresses and other network information to an interface, based on BOOTP. A device on a network with no network information can broadcast a request for an IP address, subnet mask, default gateway and other information from a DHCP server on that same network. DHCP is defined in RFC 2131.

Dynamic Routing Engine

A subset of the firewall engine that performs load balanced routing. The process that determines the gateway for a particular connection, after assessing which network link provides the fastest round trip time. Please see “*Load Balanced Routing*” on page 309.

E

Encryption

Used for data security, it translates any data into a secret code. Public-key encryption and symmetric encryption are the main types of encryption. Decrypting ciphertext (encrypted data) into plaintext requires access to a secret key. Please see “*Cryptography*” on page 298.

Encryption Key

The key, or additional data used to convert plaintext to ciphertext. In symmetric algorithms, the same key is the decryption key as well. In public key algorithms, a different, but related key is used to convert the ciphertext back into plaintext.

ESP (Encapsulating Security Payload)

A security protocol supported by IPsec protocol to enhance traffic security. It has some similar functions as AH (see “*AH (Authentication Header)*” on page 292), and enhances data privacy with the help of datagram payload encryption.

F

Firewall

A barrier or “choke point” between two or more networks, which examines, controls and/or blocks the flow of data between those networks. Often thought of as a defense between a corporate network and the Internet, firewalls can also protect internal networks from each other.

Firewall Cluster

A group of firewalls that, through clustering technology, process the work normally performed by a single firewall machine.

Firewall Engine

The application software or processes that run on a firewall, performing the actual examination and access control of data.

Firewall Node

A single device, often a specialized PC or router, that runs firewall software, and performs the functions of a firewall as part of a firewall cluster.

Firewall System

A collection of applications used to implement security policies and monitor network traffic at one or more sites. A firewall system consists of firewall engines, management servers, Log Servers and GUIs.

Forwarding

Please see “*Packet Forwarding*” on page 315.

Fragmentation/Defragmentation

The process by which a large block of data is broken up into smaller pieces (datagrams), so that it can be packaged and transmitted by the underlying network technology (fragmentation). Once the smaller pieces arrive at their destination, the datagrams are reassembled into the larger block of data (defragmentation).

FTP (File Transfer Protocol)

A TCP-based Internet standard application layer protocol for transferring files between devices. FTP is standardized as STD0009, or RFC959.

G

Gateway

A device by which users of one computer service or network can access information on a different service or network. This can be done by means of hardware devices (bridges), by computer programs that do the translation, or by both.

Granularity

It describes the degree to which something can be divided into convenient, small, independent parts.

H

Hash Signature

A cryptography-related concept that refers to a digital fingerprint associated with a given message and computed with one-way algorithms. Hash signatures are used to secure the integrity of encrypted data, ensuring that no tampering has taken place during transmission. See also “*MD5*” on page 312, and “*SHA-1*” on page 322.

Heartbeat

A protocol that the firewall nodes of a cluster use to negotiate load balancing, monitor each other, and perform other tasks that are needed for collaboration between nodes.

High Availability

The implementation of clustering technology, hot standby technology, or general redundancy in a system to increase the availability of an application, service, or network beyond what a single system is capable of providing. Increased availability is achieved by eliminating all single points of failure, with clustering technology providing the highest level of availability.

Host

For StoneGate, any device connected to a TCP/IP network, including the Internet, with one or more IP addresses. Hosts are distinguishable from gateways or routers, in that they do not forward, or route, packets to other networks.

Hot Standby Technology

A set of methods and algorithms used to implement reliable solutions where one node handles the work load with the support of a back-up node in case of failures.

HTTP (Hypertext Transfer Protocol)

Internet standard for transferring data on the World Wide Web. HTTP is defined in RFC2616, among others.

Hybrid Cryptosystem

A system using both asymmetric and symmetric encryption. Public key techniques allow key management and digital signatures which are not readily available with symmetric encryption procedures. The symmetric algorithms, however, can perform the bulk of the encryption process more efficiently than asymmetric keys.

ICMP (Internet Control Message Protocol)

Please see "*Internet Control Message Protocol (ICMP)*" on page 305.

ICMP Tracking

Information maintained by the firewall engine to group together ICMP requests and replies, handling them as a virtual connection. Please see "*Virtual Connection Tracking*" on page 328.

Identification/Identity

The unique name of a user, group or object.

IKE Proposal

In the Security Association (SA) component of an IPsec VPN, the suggested encryption algorithms, authentication methods, hash algorithms, and Diffie-Hellman information to be used. The initiator of an IPsec tunnel can make multiple proposals, but the responder only sends one proposal in return. Please see “*Internet Key Exchange (IKE)*” on page 306. Please see “*Security Association (SA)*” on page 321.

Insider Attack

An attack which originates from inside a protected network.

Intelligent Rule Base

An improved set of methods for translating corporate security policy documents into a set of rules that can be implemented by a firewall engine. Intelligent Rule Base technology implements sub-rules, templates, access controls and other objects, enabling flexible methods for:

- managing multiple firewalls that share some common policies
- managing a set of firewalls with the same high-level policy
- distributing firewall administration among multiple users

Internal Network

A term used for the networks and network resources that the firewall is protecting. A typical policy allows most outgoing connections from the internal network to other networks, such as the Internet. Most connections from the Internet to the internal network(s) are denied.

Internet Control Message Protocol (ICMP)

An extension to the Internet Protocol (IP), defined by RFC 792. ICMP supports packets containing error, control, and informational messages.

Internet Key Exchange (IKE)

A protocol defined by the IPsec protocol to securely exchange key-related information between connecting hosts. It is a subset of ISAKMP.

Internet Service Provider (ISP)

Internal networks are typically connected to the Internet by the connections leased from an ISP. A contract with an ISP typically contains the physical transmission media, modems, a customer router, IP addresses, and services such as a news or mail server.

Intrusion Detection

Determining, and being made aware of break-ins or break-in attempts, either manually or via software systems that operate on logs or other network information.

Intrusion Detection System (IDS)

A set of software and possibly hardware configured to analyse network activity and logs to determine if a break-in has occurred or is being attempted.

IP (Internet Protocol)

The network layer for the TCP/IP protocol suite. IP is a connectionless, packet switching protocol, which controls packaging messages into individual packets, routing them across one or more networks, and reassembling the packages into the original message. IP is defined in STD 0005, or RFC 0791.

IP Address

A 32-bit address (the current IPv4 standard; a newer version, IPv6 increases the space to 128-bit addresses) that identifies an interface of a particular device on a TCP/IP-based network.

IP Address Sharing

A technique where nodes of a cluster share an IP address.

IPComp (IP Payload Compression Protocol)

IP payload compression is a protocol to reduce the size of IP datagrams. This protocol will increase the overall communication performance between a pair of communicating gateways by compressing the datagrams, provided the nodes have sufficient computation power, through either CPU capacity or a compression coprocessor, and the communication is over slow or congested links. IPComp is particularly effective when encryption is applied to IP datagrams. IPComp is defined in RFC 2393.

IP Splicing (or Hijacking)

An attack performed by intercepting and using an active, established session. Often occurs after the authentication phase of the connection is complete, giving the attacker the permissions of the original, authenticated user. Encryption at the session or network layer is typically the best defense from such an attack.

IP Spoofing

A technique used to obtain unauthorized access to computers by sending connection requests with tampered headers, simulating a trusted source.

IPsec (IP Security)

A set of protocols supporting secure exchange of packets. Used for the implementation of VPNs, it provides transport and tunnel encryption modes. IPsec is defined in RFC 2401.

IPsec Proposal

Suggested encryption algorithms, hash algorithms, authentication methods, etc. to be used for an IPsec tunnel. Please see *“IKE Proposal”* on page 305.

ISAKMP (Internet Security Association Key Management Protocol)

An open-ended encoding protocol necessary for IKE negotiation when establishing Security Associations. Please see “*Security Association (SA)*” on page 321.

ISP (Internet Service Provider)

Please see “*Internet Service Provider (ISP)*” on page 306.

L

LAN (Local Area Network)

Please see “*Local Area Network (LAN)*” on page 310.

LDAP (Lightweight Directory Access Protocol)/LDAPS

A set of protocols for accessing directories of information about users and objects. Often, LDAP directories contain user information, phone numbers, e-mail addresses, etc. LDAP is a simplified, open version of the X.500 directory protocol. LDAP version 3 is defined in RFC 2251, among others. LDAPS is a secure implementation of LDAP, where directory connections are encapsulated with SSL.

Least Privilege

Designing operational aspects of a system to operate with a minimum amount of privilege. This technique reduces the authorization level required to perform tasks, decreasing the likelihood that a process or user can be forced to perform unauthorized activity.

Lifetime, soft

The interval at which the IPsec participants should begin to negotiate a replacement SA (security association). Please see “*Security Association (SA)*” on page 321.

Lifetime, hard

The interval at which the current SA for an IPsec tunnel is no longer valid. Please see “*Security Association (SA)*” on page 321.

Load Balancing

A process for distributing work evenly across multiple, available devices to avoid overwhelming any single system.

Load Balancing Filter

A software component that determines which network connections should be handled by a particular node in a cluster, based on address information, current load, performance of individual machines, and other factors.

Load Balancing NAT

The even distribution of work between multiple servers, using network address translation (NAT) to adjust the flow of traffic to each. Please see “*NAT (Network Address Translation)*” on page 313.

Load Balancing VPN

The even distribution of network traffic through multiple VPN tunnels between two gateway clusters.

Load Balanced Routing

A method through which routes to destinations are chosen after determining the fastest response time through multiple gateways. The application of multi-link technology to determine which network link provides the best round trip time. Please see “*Dynamic Routing Engine*” on page 300.

Load Sharing

The distribution of work between multiple devices. Similar to load balancing, but not as effective, since the techniques used do not ensure an *equal* distribution of the work load. Load sharing is typi-

cally a static method of distributing a load, whereas load balancing is often a dynamic method.

Local Area Network (LAN)

A data network that links two or more computers within a geographically limited area, such as an office, floor, or building.

Logs, Logging

The information stored, and the process of storing that information, about events that occurred on the firewall or networks.

Log Retention

The period of time that logs are maintained, stored and/or archived.

Log Server

A software component of the management system responsible for storing and managing log data.

Log Spool

A temporary storage area in a firewall node for log data before it is sent to a Log Server.

M

Main Mode

An IKE negotiation mode, which exchanges six messages between the end-points of an IPsec tunnel to complete the negotiation of authentication and keys. Optionally, Perfect Forward Secrecy (PFS) can be applied to protect further negotiations. Please see “*Aggressive Mode*” on page 292. Please see “*Perfect Forward Secrecy (PFS)*” on page 315.

Management Domain

A management domain represents the set of VPN sites where all roaming users are allowed to connect to some of the sites and communicate through the encrypted connections set up by the VPN configuration of the security gateways.

Management Network

The network used for communication between firewall engines, management servers, Log Servers and the GUI.

Management Server

A process responsible for communicating with the firewall engines, for example to upload a new policy, communicate with the GUI to let the administrator modify a policy, or manage the policy and other configuration data of the firewall system. Often considered the machine running the software process to perform the role of a management server.

Management System

The system consisting of management servers, Log Servers and databases that is used to manage the firewall engines, and to store and manage alerts and log data. Also includes defining the security policy and network interfaces.

Manual Encryption Keying

A set of symmetric keys that have no defined means of distribution. In other words, a person must somehow manually transfer the keys to be used, hopefully through some reasonably secure method.

Mapping

Converting data encoded in one format to another format.

Masquerading

The process of hiding an entire subnet, or set of networks, behind a single external address. Please see “*NAT, many-to-one*” on page 314.

Maximum Transmission Unit (MTU)

The largest physical size of a datagram that can be transmitted over a network without fragmentation. Often expressed in bytes, it can apply to frames, packets, cells or other media, depending on the underlying topology.

MD5

An algorithm used for keying hash functions. It generates a 128-bit signature from an input of any length. MD5 was developed by Ronald Rivest (see “*RSA (Rivest, Shamir and Adleman)*” on page 319) as an improvement to MD4, and is defined in RFC 1321. Please see “*Hash Signature*” on page 303.

Multicast

A technique by which a set of packets are sent to a group of machines sharing a common address. Unlike broadcast, it does not include all machines, and unlike unicast, it usually has more than one member of the group.

Multi-Layer Inspection

A hybrid firewall technology that incorporates the best elements of application level and network level firewalls, with additional technology to enable the secure handling of many connection types.

Multi-Link Technology

A set of techniques that are applied to connect one site to another, or to the Internet, using more than one network link (ISPs, leased lines, or other topologies). Applications of multi-link technology

include the Dynamic Routing Engine and multi-link VPNs. Please see “*Dynamic Routing Engine*” on page 300.

Multi-Link VPN (Multi-Link Virtual Private Network)

A VPN solution where a site is connected to another site by more than one network link, and these connections are used to pass VPN traffic. Compared to a traditional, single-link VPN, multi-link VPNs offer high availability and increased performance.

Multi-Route Tunnels

Multi-route tunnels refer to the logical tunnels connecting each pair of sites in a multi-link VPN. As the name suggests, these tunnels consist of a set of tunnels called routed tunnels. Please see “*Routed Tunnels*” on page 319.

N

NAT (Network Address Translation)

A mechanism for assigning local networks a set of IP addresses for internal traffic and another for external traffic. NAT was originally described in RFC 1631 as a means for solving the rapidly diminishing IP address space. It provides a supplemental security purpose by hiding internal IP addresses or to enable hosts with "invalid" (non-routable) addresses to communicate on the Internet.

NAT, dynamic

A way to translate network addresses, where for each original address, a translated address and possibly a port are selected dynamically from a predefined pool of either addresses or address/port combinations.

NAT, many-to-many

Dynamic NAT, where translated addresses and ports are selected from a pool of address-port combinations. Alternate term: 'NAT, address/port pool'.

NAT, many-to-one

Dynamic NAT, where one fixed translated address is allocated to all original addresses. A port is dynamically selected from a pool of available ports. Alternate term: 'NAT, port pool'.

NAT, static

A way to perform NAT, where for each original address, there is a single, predefined translated address.

NDI (Node Dedicated Interface)

Please see *“Node Dedicated Interface (NDI)”* on page 314.

Network Level Firewall

Often referred to as a packet filter, a network level firewall is a firewall that examines traffic at the network protocol level, and controls access based on network address information. Please see *“Packet Filtering”* on page 315.

Network Interface Card (NIC)

A hardware adapter, or card, that allows you to connect a network cable to a device.

Node

A single device that performs a fraction of a given work load, as part of a cluster. Please see *“Firewall Node”* on page 302.

Node Dedicated Interface (NDI)

A network interface with a unique IP address for each machine. NDIs can have one or more of the following roles assigned to

them: primary heartbeat/sync, backup heartbeat/sync, management, and default for outbound connections, or they can operate with no assigned role at all.

P

Packet

A unit of data sent across a network.

Packet Filtering

A method of controlling access to a network, or set of networks, by examining packets for source and destination address information, and permitting those packets to pass, or halting them based on defined rules.

Packet Forwarding

A process of receiving a packet from a network interface and sending it out through another network interface.

Packet Sniffer

Please see “*Sniffer*” on page 323.

Perfect Forward Secrecy (PFS)

A property of IKE transactions that enhances the secrecy of keys, but requires additional processing overhead. PFS ensures that the distribution of key-related information remains independent from previously existing key material. Please see “*Internet Key Exchange (IKE)*” on page 306.

PKI

Please see “*Public Key Infrastructure (PKI)*” on page 317.

Policy

Please see “*Security Policy*” on page 322.

Policy Routing

The process where routing decisions are made based on information that is not normally used in routing, such as the source IP address, port information, or service type.

Port Address Translation (PAT)

A process, similar to network address translation (NAT), where the source or destination port is changed to a different port. PAT is often used to disguise, or masquerade a service in place of another. Please see “*NAT (Network Address Translation)*” on page 313.

Pre-shared Key

A symmetric key that is exchanged with another system, by some other means, prior to the negotiation and establishment of an encrypted communication.

Pre-shared Key Authentication

The process by which two systems prove their identity to each other, where each system encrypts some unpredictable, arbitrary data with a key that has been exchanged beforehand. If they can successfully decrypt the message, it is assumed that the sender is valid.

Protocol

An agreed-upon format for transmitting data between two or more devices. Protocols typically define how to check for errors, how the sender will announce they have completed the sending of data, how the receiver will acknowledge receipt of the data, and how they will compress the data (if applicable). Please see “*Transport Protocol*” on page 327.

Protocol Agent

A process on the firewalls that assists the engine in handling a particular protocol. Protocol agents ensure that related connections

for a service are properly grouped and evaluated by the firewall engine, as well as assisting the engine with content filtering or network address translation tasks. Please see “*Connection Tracking*” on page 297.

Proxy/Proxy Firewall

A software agent that acts on behalf of a user. Typical proxies accept a user connection, determine if the user is authorized to use the proxy, and then completes the connection to a remote service on behalf of the user. Please see “*Application Layer Gateway ; Application Level Firewall*” on page 293.

Public Key Cryptography

A cryptographic system that uses two keys -- a public key, available to anyone, and a private or secret key held by only one individual. Please see “*Asymmetric Encryption*” on page 293.

Public Key Infrastructure (PKI)

A system of digital certificates, certificate authorities, and other registration authorities that verify and authenticate the validity of each party involved in a transaction. PKI's are constantly evolving, and there are currently no agreed-upon standards defining them

R

RADIUS (Remote Authentication Dial-In User Service)

A protocol for carrying authentication, authorization, and configuration information between an access server, which desires to authenticate its links, and a central, shared authentication server. The RADIUS protocol standard is currently defined in RFC 2865. A RADIUS server is a host that performs authentication for other network devices, using this protocol to communicate with those devices.

Refragmentation

A technique to fragment outbound packets from the firewall in the same manner in which they were fragmented when the firewall received them. Please see “*Virtual Defragmentation*” on page 328.

Related Connection

A connection that has a relationship to another connection defined by a service. For example, the FTP protocol defines a relationship between a control connection, and one or more data connections at the application level. The firewall may be required to allow a connection that would otherwise be discarded, if it is related to an already allowed connection.

Request for Comments (RFC)

A document that outlines a proposed standard for a protocol. RFCs define how the protocol should function, and are developed by working groups of the Internet Engineering Task Force (IETF), and reviewed and approved by the Internet Engineering Steering Group (IESG). Please see <http://www.rfc-editor.org/>.

Roaming (Roaming Access)

A feature that allows mobile users to connect remotely to their corporate network through an encrypted tunnel to one or more StoneGate security gateways. Mobile users need to have a VPN client (e.g. StoneGate IPsec VPN client, PGPnet, or Sentinel) installed and configured on their laptops.

Route

The set of routers or gateways a packet travels through in order to reach its destination. In TCP/IP networks, individual packets for a connection may travel through different routes to reach the destination host.

Routed Tunnels

Routed tunnels are the actual tunnels that are combined logically within a multi-route tunnel of a VPN. They represent all possible routes that connect the security gateways of distant sites. The individual tunnels may connect the two gateways through different network links. Please see “*Multi-Link VPN (Multi-Link Virtual Private Network)*” on page 313. See also “*Multi-Route Tunnels*” on page 313.

Router

A device that connects any number of networks, usually local area networks (LANs). Such devices examine the address information in packets arriving at the router, and determine where the packet should go next to get to its destination.

Routing

The process of determining the route of an outgoing packet. Routing decides the outgoing network interface and the next hop for the packet.

Routing Table

A database maintained on every router or gateway with information on paths to different networks.

RSA (Rivest, Shamir and Adleman)

A common asymmetric key algorithm using variable length keys (theoretically unlimited length). Generally, 1,024-bit keys or greater are used for both digital signatures and encryption.

RTT (Round-Trip Time)

The total time required for a packet or set of packets to travel from one host to another host at a different location, and return to the original host.

Rule

An expression used to define the eventual outcome of packets arriving at the firewall, which match certain conditions (e.g., source and destination address, protocol, user).

Rule Base

A set of rules that determine which communications are permitted to traverse the firewall gateway. In StoneGate they are a modular building block used to express a corporate security policy.

Rule Base, intermediate

A set of intermediate rules.

Rule Base, run-time

A set of rules that have been processed by the management system and loaded into the firewall engines. This set of rules is the set currently being used by the firewall engines to control network traffic.

Rule Option

Additional actions or features that should be applied to a packet matching a given rule. Rule options specify whether or not to log connection information or send alerts, for example. They are in addition to, and have no bearing on, the actual action taken as specified in the Action column, and rule options are not required.

S

SA (Security Association)

Please see “*Security Association (SA)*” on page 321.

Screened Host

A host on a network behind a screening router. The degree to which that host can be accessed depends on the rules in the router.

Screened Subnet

A subnet behind a screening router. The degree to which that subnet can be accessed depends on the rules in the router.

Screening Router

A router configured to permit or deny traffic based on a set of permission rules installed by the administrator.

Secret Key Cryptography

Please see “*Symmetric Encryption*” on page 325.

Security Association (SA)

A unidirectional, logical connection established for securing communications between two sites. A security association records the information required by one site to support one direction of the IPsec connection whether inbound or outbound. According to the IPsec protocol, security associations are implemented by means of AH (Authentication Headers) or ESP (Encapsulating Security Payload). It uses transport mode for communications between two hosts and tunnel mode for communication between security gateways. See also.

Security Gateway

A device, typically a firewall, that performs (IPsec) encryption/decryption on packets sent between sites via untrusted networks.

Security Parameter Index (SPI)

A value used by AH and ESP protocols to help the firewall cluster select the security association that will process an incoming packet. Please see “*Authentication Header (AH)*” on page 294. See also “*ESP (Encapsulating Security Payload)*” on page 301.

Security Policy

A set of templates, rule bases, and subrule bases that define the written corporate policies for securing network and computer resources. Security policies are defined in the rule bases through the GUI and management system and installed on firewalls, which then use the policy to determine the appropriate action to take on packets traversing the network.

Server Cluster

Please see “*Cluster*” on page 297.

Service

An application level protocol, for example FTP, HTTP or SMTP. The service is determined from the TCP/UDP port number.

Session Stealing

Please see “*IP Splicing (or Hijacking)*” on page 307.

SHA-1

A cryptographic algorithm used for hash functions. It generates a 160-bit signature from an input of any length. Please see “*Hash Signature*” on page 303.

Simple Mail Transfer Protocol (SMTP)

A protocol used to transmit e-mail over computer networks. SMTP is defined in RFC 821, among others.

Simple Network Management Protocol (SNMP)

A set of protocols for managing and monitoring complex networks. Devices supporting SNMP can communicate their status to a centralized monitoring service. SNMP is defined in RFC 1157, among others.

Single Point of Failure

The point at which the failure of a single device or component of a system will lead to either the failure of the entire system, or the inability to use services normally provided by that system. Redundant systems, using high availability technologies, eliminate single points of failure.

Site

An office facility, or similar location, in a relatively small physical area. From the firewall's point of view, a site is a set of resources protected by a single firewall cluster.

SMTP

Please see “*Simple Mail Transfer Protocol (SMTP)*” on page 322.

Sniffer

A device and/or program that captures data traveling over a network. Sniffers are often used for troubleshooting network problems, as they can show the packet flow taking place. They can also be used maliciously to steal data off a network as well.

SNMP

Please see “*Simple Network Management Protocol (SNMP)*” on page 322.

Social Engineering

An attack method by which users are deceived into revealing information about systems or security through the telephone, or by persons impersonating authorized users.

SPI (Security Parameter Index)

Please see “*Security Parameter Index (SPI)*” on page 321.

SSH (Secure Shell)

Secure Shell is a program to log into another computer over a network, to execute commands in a remote machine, and to move files from one machine to another. It provides strong authentication and secure communications over insecure channels. Often used as a replacement for insecure programs such as telnet or rsh.

State Synchronization

The communication of connection tracking information between several firewall nodes in a cluster. Please see “*Connection Tracking*” on page 297.

State Synchronization, full

State synchronization, where all connection tracking information is transferred to the other nodes of a cluster.

State Synchronization, incremental

State synchronization, where only the data of connections changed after the last synchronization are transferred to the other nodes of a cluster.

State Synchronization, on-demand

State synchronization triggered by a need to transfer connection tracking information, for example to allow a related connection through that may end up traversing another node in the cluster.

State Synchronization, timed

State synchronization triggered by a timer. Timed state synchronization may be either full or incremental.

Static Routing

A form of routing that has permanent routes between networks programmed into every routing table.

Stream Cipher

Please see “*Stream Encryption Algorithm*” on page 325.

Stream Encryption Algorithm

A symmetric cipher which can encrypt an arbitrarily sized stream of data, and generate a stream of ciphertext output.

Sub-Rule Base

A set of rules that are separated from the main rule base, based on some common category, such as the services allowed, or the destination. In this way, related rules can be grouped together to make the entire rule base easier to understand. Because subrules are only processed if the general rule in the main rule base matches, the overall processing time is improved.

Symmetric Encryption

An encryption mechanism that uses the same shared secret key for encrypting and decrypting messages. It is often referred to as symmetric bulk encryption since it processes large amounts of data rather quickly. Also known as conventional or secret key cryptography. There are two main types of symmetric encryption algorithms, bulk and stream encryption (also known as block ciphers and stream ciphers). Common symmetric algorithms are DES and 3DES. See also “*Asymmetric Encryption*” on page 293.

T**TACACS**

Terminal Access Controller Access Control System. TACACS is a simple UDP-based access control protocol, originally developed by BBN for the MILNET, that provides remote access authentication and related services, such as event logging. User passwords are administered in a central database rather than in individual routers,

providing an easily scalable network security solution. TACACS is defined in RFC 1492.

TACACS+

TACACS+ improves on TACACS and XTACACS by separating the functions of authentication, authorization, and accounting, and by encrypting all traffic between the authentication server and the device requesting authentication. It allows for arbitrary length and content authentication exchanges, which will allow any authentication mechanism to be utilized with TACACS+ clients. Additionally, TACACS+ is now a TCP-based protocol.

Takeover Period

Time interval during which the active firewall nodes collaborate to redistribute the work load of a failed node.

TCP (Transmission Control Protocol)

A connection-oriented, and stream-oriented, protocol that permits two computers to establish a reliable connection. TCP is defined in RFC 793, now published as STD 7.

TCP/IP (Transmission Control Protocol/Internet Protocol)

A commonly used term for the entire Internet protocol suite.

TOS Flags

A data field from IP headers that provides information on the type of service being used.

Traffic Management

The control, definition, and management of how packets or connections should flow through firewalls, routers, network links, VPNs or other gateway objects, based on load balancing, clusters, availability of links and more.

Transparent Proxy

A technique whereby a connection is routed to a proxy server, which then establishes a second connection to the original destination host, but the entire transaction takes place without notifying the user, or requiring the user to perform any additional actions.

Transport Protocol

Any protocol that communicates and functions on the transport layer of the TCP/IP protocol stack. These protocols function above the network layer, and are usually responsible for error correction, quality of service, and other characteristics not handled by the network layer. TCP, UDP, and IPsec are common examples of transport protocols.

Trojan Horse

A software program that appears to perform its normal function, but which in fact contains a trapdoor or attack program.

Tunneling

A technology that enables one network to send its data through another, perhaps dissimilar, network. Tunneling works by encapsulating, or packaging, a network protocol within packets carried by the second network.

Tunnels (SS, HH, SH)

HH (host-to-host) tunnels designate tunnels between two hosts. SS (subnet-to-subnet) tunnels are tunnels between subnets. SH (subnet-to-host) tunnels are tunnels between subnets and hosts.

Tunneling Router

A router, or similar system, capable of routing traffic by encrypting and encapsulating it for transmission across an untrusted network, for eventual decapsulation and decryption.

U

UDP (User Datagram Protocol)

A connectionless protocol that, like TCP, runs on top of the IP network layer. Unlike TCP, UDP provides very few error recovery services, offering instead a direct way to send and receive datagrams over an IP network.

UDP Tracking

Information maintained by the firewall engines to group together UDP requests and replies, handling them as a single virtual connection. Please see “*Virtual Connection Tracking*” on page 328.

V

Virtual Connection Tracking

Superset of UDP tracking, ICMP tracking, etc. A technology that is used by the firewall engines for connectionless network protocols like UDP and ICMP. The firewall engines keep track of virtual connections by grouping together packets that are related, based on information in the packet headers. Please see “*Related Connection*” on page 318.

Virtual Defragmentation

A procedure in which incoming packet fragments are collected. The packet is defragmented for processing by the firewall engine, and refragmented before it is transmitted again. Please see “*Fragmentation/Defragmentation*” on page 302.

Virtual Local Area Network (VLAN)

A local area network which is defined through software in a switch or other internetworking device, rather than by the more traditional hardware division.

Virus

A piece of computer code that attaches itself to another program or data file. A virus may, or may not, contain a trapdoor or other attack programs, but always replicate themselves to other systems.

Virtual Private Network (VPN)

A set of devices connected to one or more public networks, that encrypt communications amongst themselves. Effectively, the devices create a tunnel over the public network(s) as if they were connected by private lines instead.

W**Worm**

A standalone program that, when run, copies itself from one host to another, and then runs itself on each newly infected host.

Guidelines for Building Network Security

This appendix shall give some general guidelines for building solid network security solutions. It will not provide a definite reference for a perfectly secure implementation; nor will it provide an exhaustive description of all possible security considerations. Instead, it will present some important points that are good to keep in mind when building your network security with the use of firewalls. These points are proven to be noteworthy by experienced network security specialists. Your actual security implementation will, of course, depend on many variables that are characteristic to your own network and organization.

The Main Security Objectives

The main objectives for any information security policy are the three following concepts:

- confidentiality
- integrity
- availability

Confidentiality means that classified data will not be compromised in any circumstances to any unauthorized parties. The threats to confidentiality vary from e-mail privacy violation to disclosure of business critical secrets.

Integrity means that data, in storage or in transit, will not be modified by any unauthorized parties.

Availability means that the data and the resources will be continuously available to the authorized parties.

Your information security policy should aim at safeguarding these objectives. In order to achieve this goal, it is important to try to define any possible weak spots in your systems and networks, and to use any required means to fortify them. Such weak spots can be a cause of accidental damage to your invaluable business assets, and they can also be exploited by malicious intruders or attackers.

The Corporate Information Security Policy

The overall corporate information security policy consists of many different factors, data networks security being one of them. The top management of any organization need to set a clear direction to information security. They should demonstrate their support and commitment issuing a well defined information security policy across the organization.

There should be a written policy document publicly available to all members of the organization. It should include at least the following considerations:

- definition of information security, its scope, objectives, and importance
- statement of management intention, supporting the goals and principles of information security
- explanation of specific security policies, principles, standards, and compliance requirements that serve as a collection of guidelines.

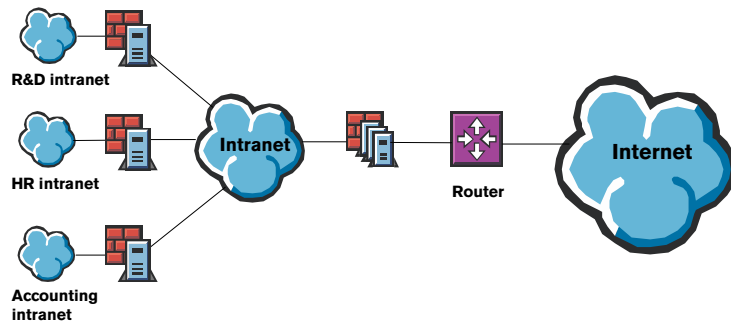
There should also be a defined process for updating and auditing the policy, in order to adapt it in the rapidly changing environment.

Firewall is a device that is used to enforce certain parts of the security policy. They are mainly used for controlling the access of information, and for ensuring the correct and secure operation of the data infrastructure. In the following sections, we will give some useful guidelines how to use firewalls in implementing network security.

Designing the Network Topology

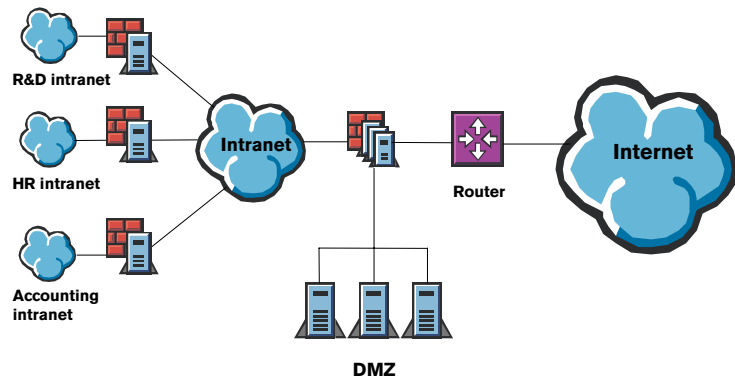
Firewalls are the key components for enforcing the network security policy. By definition, firewalls connect (and simultaneously, separate) networks on different security levels. Firewalls are used to connect internal networks to the public Internet, as well as different parts of the internal network.

All the traffic that should be subject to monitoring need to pass through at least one firewall. The more security requirements a network has, the more important it is to protect it with a series of firewalls. This will lead to a tree-like topology, where the most sensitive private networks, for example, R&D, human resources or accounting intranets, are located the furthest away from the unsecured public networks. This is a good rule of thumb for designing your network topology. Figure A.1 exemplifies a simple tree topology.

FIGURE A.1 Example of a simple tree network topology

Most organizations need to have certain publicly available services such as Web, FTP, and e-mail servers in their network. However, it is not safe to place such services in your internal network because they can fairly easily be exploited in a harmful way. Direct connections from outside to your internal network must be limited to the minimum. By default, such services should be located in a special subnetwork, *demilitarized zone* (DMZ). The connections from the Internet to the DMZ should be protected by a firewall. Moreover, the necessary connections from the DMZ to the intranet must also go through a firewall. In addition, you can define a rule that lets the servers in the DMZ accept only certain types of service requests. This will reduce the risk of attacks towards your public servers. Figure A.2 illustrates a DMZ protected by a firewall.

FIGURE A.2 *DMZ protected by a firewall cluster*



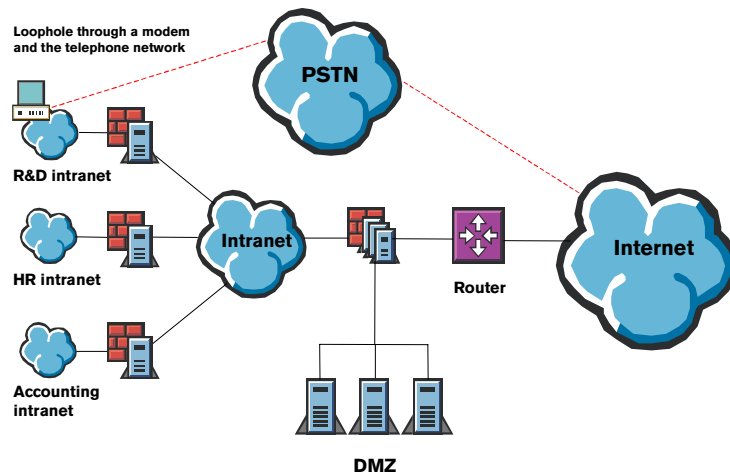
As regards Domain Name System (DNS) servers, it is advisable to have a different DNS for external and internal use. A DNS should not reveal the structure of your internal network to outside parties. To locate the publicly available DNS in a DMZ protected by a firewall is a good solution. The internal DNS should be located on a protected intranet.

Often, however, there is a need to have connections between two or more protected sites over an unsecured network. In order to provide data security in such situations, it is recommended to use Virtual Private Networks. VPN offers encryption and authentication services, so that the integrity and confidentiality of data can be preserved with a reasonable guarantee even when transmitted over a hostile public network.

When a tree network topology is implemented, it is important to ensure that the networks are connected only through firewalls. There

should be no loopholes that would offer a connection bypassing the firewalls. In Figure A.3, there is an example of a loophole that undermines the otherwise well planned and secure network topology. There is an unsecured modem connection from a computer located in an intranet, through the public telephone network (PSTN), to the Internet. This type of situations can easily go unnoticed. However, they should be eliminated.

FIGURE A.3 *A warning example of a loophole*



High Availability

Naturally, in any network topology it is of utmost importance to avoid any possible single points of failure. A breakdown or congestion of a single firewall node could block the whole network. The solution is to use redundancy protection by clustering firewalls, and preferably also other crucial network elements (such as routers, switches, critical servers, ISP links, and VPN links). The resulting high availability

together with load balancing possibility will decrease this risk significantly.

Designing the Rule Base

After discussing how topology issues affect the general network security, we shall next describe general guidelines for planning a solid rule base. Again, it is impossible to describe a complete set of rules that would guarantee your network's security. Yet we can offer some good points that should be taken into account when planning the rules. Let us first consider some general policy issues, which determine the building of a rule base.

User Authentication

The firewall rule bases without user authentication monitor and filter traffic only on the basis of IP addresses and some protocol information (protocol type, possible ports, and so on). Forging of IP addresses is not complicated, and not even anti-spoofing measures provide absolute security against IP address spoofing. The use of spoofed addresses can grant hackers access to sensitive hosts, and cause considerable damage. The introduction of user authentication in the relevant parts of the rule base will add an extra dimension to your network security. This way only trusted people can be given permission to certain types of connections.

Inbound and Outbound Connections

It is important to bear in mind that accepting a connection from the public network or opening an outbound connection from the internal network, contains a potential data security risk to your network. For example, in an open TCP/IP connection, there is a data flow in both directions, and this can be exploited for harmful intentions.

The use of firewalls with well-crafted rule bases to determine whether an opened connection is permissible reduces the hazards. Additional

protection can be gained by using content screening servers. Nevertheless, the risks cannot be eliminated totally. Thus, the enforcers of the security policy must be wary of not having a false sense of security. A successful security policy requires continuous maintenance and vigilance.

To allow connections from the Internet to your internal network is always potentially hazardous, because of Trojan Horses and viruses, for example. Therefore, the inbound connections should be kept to the minimum. The few types of inbound connections that are truly necessary should be defined very accurately. The permitted protocols should be the ones known to be safe and easy to control. In order to avoid direct connections from outside to your network, it is recommended to pass the inbound traffic via a firewall protected DMZ (see Figure A.2 on page 335).

As regards the outbound connections, a strict security policy will increase the administrative workload, if a specific permission must be applied for most outbound connections. On the other hand, a loose policy will generate less administrative work, while it will possibly allow, for example, the use of unsecured peer-to-peer programs. It is good to remember, however, that whenever outbound connections are allowed, it is always possible that some malicious parties would try to exploit them. For example, if a harmful agent program gets inside your network, it can use the outbound connections for smuggling confidential data to an information thief.

There is no one right answer for determining which is a tolerable level of risks involved in connections; it has to be judged case by case. A well designed firewall rule base combined with proper education of network users in good security practices should, however, diminish the security risks caused by opened connections.

General Rule Base Design Guidelines

A firewall without a well designed rule base is practically useless from the network security point of view. If the rules can be bypassed easily, the firewall will lose most of its usefulness.

It is advisable to start the designing of the rule base from the rules relating specifically to the most sensitive and most accurately defined destinations, such as single hosts and firewalls. Then you can continue defining rules concerning less sensitive and larger network objects, such as groups and entire networks.

A non-detailed abstraction of the order of rules is given in Table A.1. Please note that this is merely an example, not a norm.

TABLE A.1 *Rule order outline*

Source	Destination	Service	Action	Explanation
The permitted IP addresses (such as the Management Server's or Superuser's)	The IP addresses of the Firewalls	Any	Allow	Define all the allowed connections to firewall nodes.
Any	The IP addresses of the Firewalls	Any	Discard	The drop rule for any other connections to firewalls' interfaces not specified by previous rules.
The permitted IP addresses (such as the firewalls' or Superuser's)	Management Server	Any	Allow	Define all the allowed connections to the firewall Management Server
Any	Management Server	Any	Discard	The drop rule for any other connections to the Management Server not specified by previous rules.
The specifically allowed IP addresses (such as administrators' or specifically permitted networks')	For example, Authentication Server, CIS, DMZ, intranets.	Carefully defined permitted services.	Allow	Define all the allowed, well defined, connections to other sensitive targets (such as authentication servers, content screening servers, sensitive intranets, other internal networks, and DMZ networks).

TABLE A.1 *Rule order outline (Continued)*

Source	Destination	Service	Action	Explanation
Any	For example, Authentication Server, CIS, DMZ, intranets.	Any	Discard	A drop rule for any unspecified connections to each of these targets should be defined.
The permitted internal network's IP addresses	Any	Carefully defined permitted services.	Allow	Define all the allowed connections from internal networks to the public networks (Internet).
Any	Any	Any	Discard	The drop rule for every connection that has not been specifically defined above.

Logs and Alert Notifications

An important part of a network security policy is the monitoring of connections with logging. In case something hazardous actually occurs, there cannot be too much log data to help in finding out what is happening or what has happened.

One major concern in logging is, of course, the vast amount of data produced. It is not practical to store all logs, regardless of their significance. With a properly defined log filter the amount of data can be reduced to a reasonable level without compromising the efficiency of network security monitoring.

The same applies also for alert notifications; it is important that the alert notification threshold is defined to such a level that the number of generated alerts stays manageable. Too many alerts is not desirable, but nor is too few alerts. You will have to discover the golden mean also in this issue.

Conclusions

This appendix has discussed the role of the firewall in the overall scheme of information and network security policy. Undeniably, the

firewall is an essential component in that sense, but it is not the only one. You must bear in mind that also firewalls have their limitations. The firewalls require continuous maintenance and monitoring. Your security policy and the rule bases should be adjusted according to the changing requirements in the network environment. Nevertheless, careful planning of rule bases, and the use of clustering make firewalls powerful network security devices.

Multicasting

This appendix describes the general principles of multicasting, and the way it differs from broadcasting and unicasting as a transmission technique. It discusses the main features of IP multicasting. It also gives an overview of Ethernet multicasting and how it is used with Stonesoft's clustering products.

The General Features of Multicasting

Multicasting differs in certain important respects from unicasting and broadcasting as a transmission technique. A distinction can be made between multicasting traffic at the network layer (based on special class D IP addresses) and at the data link layer (based on multicast MAC addresses). The general differences how multicasting can be distinguished from unicasting and broadcasting are highlighted below.

Multicasting vs. Unicasting

In unicasting, the transmitted datagrams are intended only for a single host having a unique address. In multicasting, the data is transmitted likewise to a single address (i.e., the multicast group address), but the actual data reaches all the hosts that belong to the group identified by the multicast address in question. This way the data needs only be sent once, and not separately to each host. This naturally saves bandwidth.

Multicasting vs. Broadcasting

In broadcasting, the data is sent from a host to each other host within a given network, and they will all need to use their resources to process the data. In contrast, in multicasting, those hosts that do not belong to a multicast group won't have to use their resources for multicast data. Moreover, multicasting is not restricted to a single network, and hosts on remote networks may receive IP multicast datagrams providing that they belong to a specific host group, and that there are multicast routers forwarding the traffic. Thus, IP multicasting can in principle be used globally whereas broadcasting is limited to a single network.

IP Multicasting Overview

IP multicasting is defined in the RFC 1112 as the transmission of an IP datagram to a group of hosts identified by a single IP destination address. In addition to this common *multicast group address*, the hosts of the group all have separate and unique unicast addresses. The actual multicast host group may consist of any number of hosts, possibly even located in different networks. The number may vary over time, as hosts can join in and leave from a group at any time. Moreover, a particular host may belong to several groups simultaneously.

The multicast group addresses are class D addresses. They are identified by the high-order initial four bit sequence *1110*. In the dotted decimal notation, the multicast group address range runs from 224.0.0.0 to 239.255.255.255. There are certain special addresses:

- 224.0.0.0 is never assigned
- 224.0.0.1 is assigned to the permanent group of all hosts, including gateways, in the local network.
- 224.0.0.2 is assigned to all local multicast routers



.....

Note: *There is no multicast address that would cover each and every host connected to the Internet.*

.....



.....

Note: *Multicast IP addresses are not allowed to be used as source addresses. A multicast source address implies forging of a IP address.*

.....

The multicast groups are either permanent or transient. The permanent groups have administratively assigned IP addresses, while the addresses of the transient multicast groups can be assigned dynamically from the pool of multicast addresses not reserved for permanent groups. The IP address of an established permanent group will persist even if the group would not have any members at a given time. The transient groups will cease to exist as soon as they have no member hosts any more, and the assigned multicast address is released.



.....

Note: *Please see, for example, <http://www.iana.org/assignments/multicast-addresses> for a list of addresses registered with IANA (Internet Assigned Numbers Authority).*

.....

Multicasting Applications

On the basis of the comparisons above, multicasting may be considered a viable option for many types of transmissions. Multicasting is widely used in local area networks for various

purposes. Moreover, multicasting can be used both for receiving a publicly transmitted session on an intranet, or for transmitting an internal communication to a public network (e.g., for announcing a product launch). Multicasting is particularly important solution for bandwidth-intensive applications, such as multimedia. The most typical protocol for multicast traffic is UDP.

Multicasting may be a suitable solution, for example, for the following applications:

- work groups, electronic whiteboards
- video/voice-over-IP conferences
- real-time streaming media (such as Internet radio)
- file transfer
- spreading of any information to certain selected destinations.

Internet Group Management Protocol (IGMP)

Internet Group Management Protocol (IGMP) is an integral part of Internet Protocol. The IGMP messages are encapsulated in IP datagrams. IGMP is used both between hosts and multicast routers, and between multicast routers. It keeps multicast routers informed of the multicast group memberships on a given local network. Each host supporting multicasting must join the multicast group with the address 224.0.0.1 on each network interface at initialization time. They shall remain members of this group for as long as they are active. With IGMP, the hosts located on a LAN can inform the routers that they want to be able to receive multicast messages from external networks.

Membership Messages

Multicast routers use IGMP for enquiring periodically which multicast groups have members in the connected local networks. This is carried out by sending *Host Membership Query* messages to the all-hosts address

224.0.0.1. The hosts receiving the query respond by sending *Host Membership Reports* to all neighboring multicast routers.

A host joining a new group will immediately transmit a report, instead of waiting for a query. A host willing to stop receiving a multicast transmission will send a *Leave Report* message with the destination address 224.0.0.2 to all the subnet routers. A router receiving a *Leave Report* message will send in response a *Group Specific Query* to the multicast address in order to check whether there still are hosts in that group. If there is no response, the multicasting to that address is stopped.

Ethernet Multicasting

Above, we have discussed how multicasting is implemented at the network layer and how multicast IP addresses differ from other types of IP addresses. In addition to that, we must also distinguish multicasting at the data link layer where stations are identified by their Media Access Control (MAC) addresses, in addition to their network level IP addresses. The features of multicast as opposed to unicast and broadcast addresses described in relation to IP addressing apply also at this level.

Most local area network (LAN) topologies allow for multicasting by using some kind of a group addressing scheme. Some topologies offer better support for multicasting than others. In Ethernet (as defined in IEEE 802.3), all MAC addresses which have the least significant bit of the most significant byte as “1” are multicast addresses. Thus, for example, 01:00:00:00:00:00 and 49:aa:bb:cc:dd:ee are both multicast MAC addresses; while 02:00:00:00:00:00 and fe:fe:fe:fe:fe:fe are not. The devices with a given multicast MAC defined are able to listen to all traffic sent to that particular MAC address.



.....

Note: A specific subset of MAC addresses is reserved for mapping the IP multicasting addresses to data link layer addresses. In Ethernet, the multicast MAC addresses that correspond to multicast IP addresses range from 01:00:5e:00:00:00 to 01:00:5e:7f:ff:ff.

.....

Multicasting and StoneGate

After distinguishing between network layer multicasting and data link layer multicasting, we can now have a look at how Stonesoft's StoneGate high availability firewall uses multicasting and unicasting.

When using clustering technology, the clustered firewall nodes share a common *unicast* IP address at each *Cluster Virtual Interface* (for short, CVI). This shared IP address is assigned to those interfaces that handle traffic between networks that is to be distributed and load-balanced between all nodes, as opposed to the type of traffic the end-point of which is one specific node only. This node-specific traffic (such as management connections) is handled by the *Node Dedicated Interfaces* (NDIs). Due to these CVIs, the cluster is seen by any other network devices as a single virtual entity, rather than a group of individual nodes. Traffic addressed to this kind of interface is load-balanced between the nodes by the cluster's load balancing filter. The load balancing filter is in charge of distributing specific connections between individual nodes. Only one node in a cluster handles a given connection addressed to the cluster's common IP address, while the other nodes will ignore it.

In addition to the shared unicast IP address, each node must also share a data link layer address (MAC) at the CVI. Only this way will each of the nodes be delivered the exact same traffic. There are basically three different options for this cluster-wide MAC address,

the selection of which depends on the features of the other connected networking devices, such as switches and hubs. This document will not act as a definitive reference for different types of switch configurations, but it will give an overview of possible considerations when implementing StoneGate firewall clusters in different types of network environments. The three MAC configuration options are presented below.

Unicast MAC

A common unicast MAC can be defined at the CVIs if the cluster is connected to hubs or switches that can forward frames with a unicast destination to multiple ports. This way the network devices will forward the same packets to each of the connected firewall nodes sharing this combination of unicast IP and MAC addresses. This mode is recommended whenever the networking devices support sending packets to a specified unicast MAC address to a predefined set of ports at the same time (as opposed to one port, which is typically the default). Hubs by default support this; with switches this, however, is not so frequent, and they usually need some additional configuration. With unicast MAC, only the switches directly connected to the cluster need some special configuration.



Note:

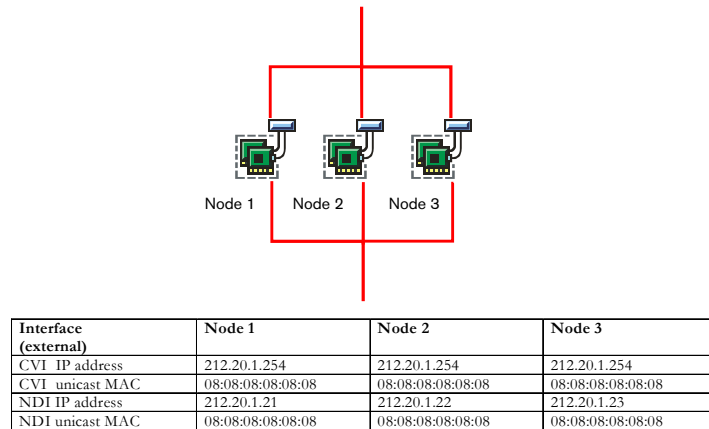
.....

In addition to the common CVI IP address, each node may optionally have also unique unicast IP addresses defined at the same physical interface as the CVI. These unicast IP addresses are assigned to Node Dedicated Interfaces (NDI), and used when an individual node is the end-point of a connection. Since there can only be one unicast MAC address at a given interface, also the node-specific NDI IP addresses are mapped to the common unicast MAC.

.....

Figure B.1 shows as an example the IP and MAC address configuration of a cluster's interfaces that are connected to an external network. By default, the CVIs of each node share one unicast IP address. The CVI of each node is mapped to a common unicast MAC address. In addition, for each node, an NDI is defined at the same physical interface as the CVI. The NDI IP addresses of each node are unique, but they all are mapped to the same unicast MAC as the CVI IP address. This is because there can be only one unicast MAC defined for a physical interface. Traffic directed from the Internet to the cluster's external CVI IP address is sent by the connected switch or hub to all of the nodes, due to the fact that they all are identified by the same unicast MAC.

FIGURE B.1 CVI with unicast MAC



Note: Unlike multicast MAC addresses, there can be only one unicast MAC address defined for each physical interface.

Multicast MAC

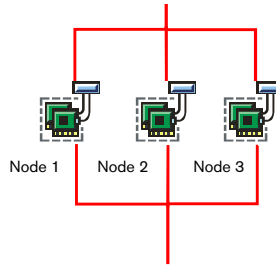
In case it's not possible to use a switch that works in unicast mode with clusters, a shared multicast MAC can be defined for the cluster nodes. Most switches support this mode, but in this mode, all switches in the same virtual LAN (VLAN) need to be configured. By default, most switches send packets with a multicast MAC address to all ports connected to the same VLAN. If the size of the VLAN is small, this type of flooding is acceptable. However, with larger VLANs, performance problems may occur as the device will need to send each packet to each port connected to the same VLAN. In some switches it's possible to prevent this by statically restricting multicast traffic with a given MAC address to some predefined ports only.



.....
Note: Some networking devices discard ARP replies specifying a multicast MAC. In that case, static ARP entries must be used.
.....

Figure B.2 presents an example where a common multicast MAC is configured for all the cluster nodes. For instance, if there is a switch not capable of sending packets with the same unicast MAC to multiple ports, this type of configuration can be used. Each of the nodes also have unique unicast MAC addresses which are mapped to the corresponding IP addresses defined at the NDIs.

FIGURE B.2 CVI with multicast MAC



Interface (external)	Node 1	Node 2	Node 3
CVI IP address	212.20.1.254	212.20.1.254	212.20.1.254
CVI multicast MAC	09:08:08:08:08:08	09:08:08:08:08:08	09:08:08:08:08:08
NDI IP address	212.20.1.21	212.20.1.22	212.20.1.23
NDI unicast MAC	04:08:08:08:08:08	06:08:08:08:08:08	08:08:08:08:08:08

Multicast MAC with IGMP

Internet Group Management Protocol (IGMP) can be used in combination with multicast MAC addresses to avoid flooding with switches that don't support statically defined destinations for multicast. In this mode, the switches are configured to send multicast traffic only to those ports from which they have received IGMP *Host Membership Report* messages corresponding to the MAC address used. Multicast with IGMP must be selected as the mode for the cluster, and IGMP snooping must be enabled on the switch. For the IGMP messaging, you need to specify a common multicast *IP address* for the cluster nodes. The multicast *MAC address* is then computed automatically from it. Do note, however, that the CVIs are still identified solely by the common *unicast IP address*; the multicast IP address is only used as the source address for the IGMP messages sent to the switch.



Note:

.....

Some routers using router redundancy protocols such as HSRP or VRRP are listening to all multicast traffic in addition to the routing related multicasting traffic. Thus, multicast packets are rerouted to the network. To prevent this, you can either configure the router to send this traffic only to the cluster ports or define the router's access control list (ACL) to drop all incoming packets with cluster's multicast MAC.

.....

Note that these three different modes presented above can be used simultaneously at a StoneGate cluster. If required, a different scheme can be configured at the CVIs for each different network (e.g., internal, external, and DMZ) connected to StoneGate.

Considering the vast number of different kinds of switches on the market, and the even greater number of different software versions running in them, we cannot provide a definite guide on how to configure all possible types of switches and routers. For further reference on different types of configurations, please visit <http://www.stonesoft.com>.

References

This appendix contains lists of references where the reader can obtain more information on the topics and technologies related to the StoneGate firewall product. Each section is grouped by general StoneGate topics in alphabetical order. Books, Web sites, and other resources for additional information or reading can be found here. The lists contained in this chapter are by no means comprehensive; other sources may be available that are equally useful.

Authentication

For more information about authentication and authentication services, we suggest:

RSA Security
<http://www.rsasecurity.com/>

CRYPTOCARD
<http://www.cryptocard.com/>

Cryptography

The following books provide useful information about cryptography and cryptographic systems.

Knudsen, Jonathan. *Java Cryptography*. 1st Ed. Sebastopol, CA: O'Reilly and Associates, 1998.

Schneier, Bruce. *Applied Cryptography: Protocols, Algorithms, and Source Code in C*. 2nd ed. New York: John Wiley and Sons, 1995.

Firewalls

For information on firewalls and related devices, we suggest the following resources.

Chapman, Brent D., et. al. *Building Internet Firewalls*. 2nd ed. Sebastopol, CA: O'Reilly and Associates, 2000.

Curtin, Matt and Ranum, Marcus. *Internet Firewalls: Frequently Asked Questions*. On-line: <http://www.interhack.net/pubs/fwfaq/>

General

The following are general sources of information and resources on networking and security.

Cooper, Alan, and Paul Saffo. *The Inmates Are Running The Asylum: Why High Tech Products Drive Us Crazy And How To Restore The Sanity*. 1999.

Garfinkel, Simon, et. al. *Practical UNIX and Internet Security*. 2nd ed. Sebastopol, CA: O'Reilly and Associates, 1996.

Hall, Eric. *Internet Core Protocols: The Definitive Guide*. Sebastopol, CA: O'Reilly and Associates, 2000.

The Security Portal for Information System Security Specialists
<http://www.infosyssec.net/>

Internet Standards

The following are sources of information on Internet standards.

Internet Assigned Numbers Authority

<http://www.iana.org/>

Internet Engineering Task Force (IETF)

<http://www.ietf.org/>

Request For Comments (RFC)

<http://www.rfc-editor.org/>

LDAP

Howes, Timothy, et. al. *Understanding and Deploying LDAP Directory Services*. MacMillan Network Architecture and Development Series, 1999.

Multicast

Goncalves, Marcus, and Kitty Niles. *IP Multicasting, Concepts and Applications*. 1998.

Organizations

This section is a list of organizations involved in network security, Internet standards, and network administration.

Computer Emergency Response Team (CERT)

<http://www.cert.org/>

Internet Engineering Task Force (IETF)

<http://www.ietf.org/>

SANS

<http://www.sans.org/>

System Administrators Guild (SAGE)
<http://www.sage.org/>

USENIX
<http://www.usenix.org/>

World Wide Web Consortium (W3C)
<http://www.w3c.org/>

TCP/IP

The following sources are recommended resources for more information on TCP/IP and related protocols.

Comer, Douglas. *Internetworking with TCP/IP Volume I: Principles, Protocols, and Architecture*. 3rd ed. Englewood Cliffs, NJ: Prentice Hall, 1995.

Hunt, Craig. *TCP/IP Network Administration*. 2nd ed. Sebastopol, CA: O'Reilly and Associates, 1998.

Virtual Private Networks

The following are recommended for more information about virtual private networks (VPNs).

Scott, Charlie, et. al. *Virtual Private Networks*. 2nd ed. Sebastopol, CA: O'Reilly and Associates, 1998.

Kaufman, Elizabeth, and Andrew Newman. *Implementing IPsec, Making Security Work on VPNs, Intranets, and Extranets*. John Wiley & Sons, Inc., 1999.

Index

A

access control list, 6

access rules, 87

- action, 89
- authentication, 90
- options, 90
- rule tag, 90
- service, 89
- source and destination, 88
- time, 90
- users, 89

actions

- allow, 89
- apply VPN, 89
- continue, 89
- discard, 89
- enforce VPN, 89
- jump, 89
- refuse, 89

address range, 86

administrative operations, 113

administrator, 111

administrator levels, 49

Administrator Manager, 58

AH (Authentication Header protocol), 163

- combining with ESP, 165

alert notification, 340

Alert Notification Man-

ager, 59

algorithms

- digital signatures, 177
- message digest, 177
- symmetric, 176

alias, 86

anti-spoofing, 74

- Anti-spoofing View, 74

architecture

- benefits of, 32
- StoneGate, 24

asymmetric encryption, 155

Audit Manager, 60

authentication, 14, 133

- client initiated, **143**
- digital signature, **159**
- firewall initiated, **141**
- one-time password, **145**
- out-of-band authentication, **140**
- token card, **145**
- two-factor authentication, **136**

authentication server, 83**authorization, 133**

- client IP, **142, 143**
- current connection, **143**
- upcoming connection, **144**

B**BGP (Border Gateway Protocol), 38****C****certificate, 31, 161****Certificate Authority, 162****certificate request, 182****CIS (content inspection server), 15****clustering**

- firewall nodes, **36**

configuration files, 31**content screening, 15****Control Panel, 54**

- StoneGate, **26**

CRL (certificate revocation list), 162**cross-reference check, 113****cryptography**

- asymmetric encryption, **155**
- digital certificate, **31**
- overview, **152**
- public key, **31**
- symmetric encryption, **153**

D**database engine, 29****destination port translation, 123****Diffie-Hellman, 157, 178****digital certificate, 31****digital signature, 156**

- data integrity, **159**
- non-repudiation, **159**

DMZ (demilitarized zone), 5, 334**DNS server, 84****dynamic IP, 171****dynamic source NAT, 122****E****Editor, 49, 108****ESP (Encapsulating Security Payload protocol), 165**

- combining with AH, **165**

expression, 85**external LDAP server, 139****F****Filtering Profile Manager, 59, 101****FIPS mode, 176****firewall**

- access control, **3, 13**
- advantages of, **4**
- authentication, **14**
- central management, **18**
- cluster, **85**
- content screening, **15**
- engines, **30**
- functions of, **13**
- high availability, **16**
- human factors, **18**
- internal attacks, **19**
- logging, **13**
- monitoring, **13**

- Multi-Layer Inspection, 11
 - NAT, 13
 - packet filter, 6
 - proxy firewall, 8
 - requirements of, 16
 - routing, 68
 - scalability, 17
 - single, 85
 - stateful inspection, 9
 - technologies, 6
 - test subsystem, 30
 - throughput, 17
 - VPN, 14
 - weaknesses of, 18
- firewall cluster, 85
- ## G
- grant list, 112
- granted elements, 110
- group, 85
- GUI (graphical user interface)
- StoneGate, 25, 54
- ## H
- hash function, 159
- high availability, 35, 336
- connections, 38
- host, 82
- HSRP (Hot Standby
- Routing Protocol), 38
- ## I
- IKE (Internet Key Exchange), 166
- phase 1, 167
 - phase 2, 169
 - SA, 167, 179
- inbound traffic management, 43
- installation overview, 61
- intelligent rule base, 81
- IP spoofing, 72
- IPsec, 162
- manual IPsec, 170
 - SA, 169, 179
- ## K
- key exchange, 156
- ## L
- Launchpad, 26, 54
- LDAP (Lightweight Directory Access Protocol), 30, 137
- LDAP server, 83
- License Manager, 60
- load-balanced routing, 39
- Log Browser, 60, 99
- Log Data Manager, 59
- Log Pruning Filter Manager, 60, 101
- log server, 29, 83, 97
- purpose of, 29
- logging, 13, 95, 340
- filters, 97
 - options, 97
- ## M
- management server, 27
- management system
- database engine, 29
 - log server, 29
 - management server, 27
 - StoneGate, 27
 - user directory, 30
- man-in-the-middle attack, 158
- MD5, 146, 163, 164
- message digest, 159

monitoring system, 28

multicast, 77

Multi-Layer Inspection, 11, 45

Multi-Link Technology, 38
benefits, 40

Multi-Link VPN, 41

N

NAT

outbound load balancing, 128

NAT (network address translation), 13, 120

destination port translation, 123
destination translation, 123
dynamic source translation, 122
hide NAT, 122
static source translation, 120

NAT rules, 91

NAT method, 92
service, 92
source and destination, 91

NAT Rules Editor, 124

NetLink, 84

NetLink pool, 84

network, 85

"Any Network", 85

Network Element Manager, 58

network elements, 82

O

one-time password, 145

Operator, 49, 109

OSPF (Open Shortest Path First), 69

outbound load balancing, 128

outbound traffic management, 41

P

packet examination, 47

packet filter, 6

packet flow, 47

path MTU discovery, 180

permissions, 111

PFS (Perfect Forward Secrecy), 178

PKI (public key infrastructure), 137, 158

policy routing, 75

pre-shared key, 178

private key, 155, 157

Protocol Agent, 11, 46, 130

proxy ARP, 129

proxy firewall, 8

public key, 155, 157

public key encryption, 155

R

RADIUS, 145

RIP (Router Information Protocol), 69

router, 82

routing protocols, 69

Routing View, 70

-
- rule base
 - design guidelines, 337
 - rule tag, 90
 - S
 - SA (security association), 163
 - IKE, 167, 179
 - IPsec, 169, 179
 - SAD (security association database), 163
 - SecurID card, 137
 - security gateway, 170
 - security policy
 - corporate, 332
 - main goals, 3, 331
 - risks, 338
 - Security Policy Manager, 58
 - server, 83
 - server pool, 84
 - Services Manager, 59
 - SHA-1, 163, 164
 - simple elements, 109
 - single firewall, 85
 - single point of failure, 4, 336
 - SPI (security parameter index), 163
 - SSL (Secure Sockets Layer), 31
 - stateful inspection firewall, 9
 - static destination NAT, 123
 - static IP multicast routing, 77
 - static routes, 68
 - static source NAT, 120
 - status display, 54
 - StoneGate
 - architecture, 32
 - communications, 31
 - Control Panel, 26, 54
 - database engine, 29
 - firewall engines, 30
 - general architecture, 24
 - GUI, 25, 54
 - Launchpad, 26
 - log server, 29
 - management server, 27
 - management system, 27
 - monitoring system, 28
 - routing, 70
 - test subsystem, 30
 - Superuser, 49, 107
 - symmetric encryption, 153
 - T
 - TACACS+, 146
 - test subsystem, 30, 31
 - traffic handlers, 84
 - U
 - upgrading
 - bandwidth, 40
 - user authentication, 337
 - User Manager, 58
 - V
 - virtual LAN, 33
 - VPN, 171, 176, 180
 - VPN (virtual private network), 14, 149
 - connection, 174
 - end-point, 170
 - multi-link, 41
-

site, **172**
site-to-site, **14**
transport mode, **164**

tunnel, **174**
tunnel lifetime, **179**
tunnel mode, **164**

VPN Manager, **59, 170**